QUANTUM LEAP
INNOVATIONS

November 10, 2011

CDR Joseph Cohn
Program Officer, Code 341
Office of Naval Research
875 North Randolph Street
Arlington, VA 22203-1995
joseph.cohn@navy.mil

RE: Contract N00014-09-C-0033 - Final Report

Dear CDR Cohn,

Work under contract N00014-09-C-0033 has been completed. Attached please find our Final Report and the SF-298 Report Documentation Page for:

Integrated Warfighter Biodefense Program (IWBP) – Next Phase

Covering the period March 2009 - October 2011

I will provide 2 CD’s containing the software (CLIN 0002) developed for this contract via Federal Express.

Thank you for your assistance on the above noted program. Copies have been distributed as per the Contract Data Requirements List – Instructions for Distribution. Since the Final Report exceeds 30 pages, a hardcopy of the report will be mailed to the Director, Naval Research Lab as per the Instructions for Distribution.

Sincerely,

Frank Abbott

Frank T. Abbott
VP of Finance, CFO
fta@quantumleapinnovations.com

cc: Dr. Ganesh Vaidyanathan, Project Manager, QLI gv@quantumleapinnovations.com
Administrative Contracting Officer – Stanley Brown, stanley.brown@dcma.mil
Director, Naval Research Lab, Attn Code 5596, reports@library.nrl.navy.mil
Defense Technical Information Center, tr@dtic.mil

REPORT DOCUMENTATION PAGE

Form Approved
OMB No. 0704-0188

1. REPORT DATE *(DD-MM-YYYY)*
11-10-2011

2. REPORT TYPE
Final Report

3. DATES COVERED *(From - To)*
March 2009 - Oct 2011

4. TITLE AND SUBTITLE
Integrated Warfighter Biodefense Program (IWBP) - Next Phase

5a. CONTRACT NUMBER
N00014-09-C-0033

5b. GRANT NUMBER

5c. PROGRAM ELEMENT NUMBER
0603729N

6. AUTHOR(S)
Abbott, Franklin T.
Vaidyanathan, Ganesh

5d. PROJECT NUMBER

5e. TASK NUMBER

5f. WORK UNIT NUMBER

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)
Quantum Leap Innovations, Inc.
3 Innovation Way, Suite 100
Newark, DE 19711-5456

8. PERFORMING ORGANIZATION REPORT NUMBER
QLI-TR-11-003

9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)
Office of Naval Research
ONR Code 341
875 North Randolph Street
Arlington, VA 22203-1995

10. SPONSOR/MONITOR'S ACRONYM(S)
ONR

11. SPONSORING/MONITORING AGENCY REPORT NUMBER

12. DISTRIBUTION AVAILABILITY STATEMENT
Distribution A: Approved for public release; distribution is unlimited. 10 November 2011.

13. SUPPLEMENTARY NOTES

The aim of the Integrated Warfighter Biodefense Program (IWBP) is to develop innovative technology that can be deployed to prevent U.S. armed forces from becoming battle or non-battle casualties, and especially to reduce morbidity and mortality throughout the increasingly complex battlespace of current operations. In this summary of the next phase of work on IWBP we report the continued development of novel software that provides a simulation environment for modeling mild Traumatic Brain Injury (mTBI), as an example of a chronic disease of great interest to the military medical community. In addition, we report on conceptual advances of our Pattern Based Discovery platform to deal with unsupervised pattern discovery. This capability is critical for gaining situational awareness in environments where an a priori target cannot be defined. Next, we discuss Optimizing Multi-Ship, Multi-Mission Operational Planning that also serves as a template for other complex operational planning scenarios, and provides a natural fit for QLI Modeling, Simulation and Optimization technologies. Finally, we conclude with a summary description of the development of a reliable model for cognitive readiness assessment using multiple modalities of sensors and the adaptation of the model from groups to individuals in specific tasking environments. In this work, we combined subjective ratings, user profiles, ECG and eye tracking to predict the user workload and other metrics of assessing readiness of warfighters within a time window of seconds.

15. SUBJECT TERMS
Biological Defense, Traumatic Brain Injury, Force Transformation, Situational Awareness, Pattern Based Discovery, Command and Control, Operational Planning, Simulation and Optimization, Cognitive Readiness

16. SECURITY CLASSIFICATION OF:			**17. LIMITATION OF ABSTRACT** UU	**18. NUMBER OF PAGES** 34	**19a. NAME OF RESPONSIBLE PERSON** Dr. Ganesh Vaidyanathan
a. REPORT Unclassified	**b. ABSTRACT** Unclassified	**c. THIS PAGE** Unclassified			**19b. TELEPONE NUMBER (*Include area code)*** 302-894-8044

UNCLASSIFIED

Quantum Leap Innovations, Inc.
Delaware Technology Park
3 Innovation Way, Suite 100
Newark, DE 19711

QLI-TR-11-003
October 2011

Integrated Warfighter Biodefense Program (IWBP) – Next Phase

Final Report

ONR Contract N00014-09-C-0033

Sponsored by the Office of Naval Research, Arlington, VA 22203

UNCLASSIFIED

Abstract

The aim of the Integrated Warfighter Biodefense Program (IWBP) is to develop innovative technology that can be deployed to prevent U.S. armed forces from becoming battle or non-battle casualties, and especially to reduce morbidity and mortality throughout the increasingly complex battlespace of current operations. In this summary of the next phase of work on IWBP we report the continued development of novel software that provides a simulation environment for modeling mild Traumatic Brain Injury (mTBI), as an example of a chronic disease of great interest to the military medical community. In addition, we report on conceptual advances of our Pattern Based Discovery platform to deal with unsupervised pattern discovery. This capability is critical for gaining situational awareness in environments where an a priori target cannot be defined. Next, we discuss Optimizing Multi-Ship, Multi-Mission Operational Planning that also serves as a template for other complex operational planning scenarios, and provides a natural fit for QLI Modeling, Simulation and Optimization technologies. Finally, we conclude with a summary description of the development of a reliable model for cognitive readiness assessment using multiple modalities of sensors and the adaptation of the model from groups to individuals in specific tasking environments. In this work, we combined subjective ratings, user profiles, ECG and eye tracking to predict the user workload and other metrics of assessing readiness of warfighters within a time window of seconds.

Contents

List of Figures

1. SUMMARY

Executive Summary

This final technical report summarizes Quantum Leap Innovations' (QLI) accomplishments with the Integrated Warfighter Biodefense Program (IWBP) through the contract close date of October 15, 2011 on ONR Contract N00014-09-C-0033.

Summary of Accomplishments

a. Extension of Gryphon platform to mTBI
b. Continued Development of Pattern Based Analytics Platform
c. Optimizing Multi-Ship, Multi-Mission Operational Planning
d. Medical Modeling for Cognitive Readiness

In this summary of the next phase of work on IWBP we report the continued development of novel software that provides a simulation environment for modeling mild Traumatic Brain Injury (mTBI), as an example of a chronic disease of great interest to the military medical community. In addition, we report on conceptual advances of our Pattern Based Discovery platform to deal with unsupervised pattern discovery. This capability is critical for gaining situational awareness in environments where an a priori target cannot be defined. Next, we discuss Optimizing Multi-Ship, Multi-Mission Operational Planning that also serves as a template for other complex operational planning scenarios, and provides a natural fit for QLI Modeling, Simulation and Optimization technologies. Finally, we conclude with a summary description of the development of a reliable model for cognitive readiness assessment using multiple modalities of sensors and the adaptation of the model from groups to individuals in specific tasking environments. In this work, we combined subjective ratings, user profiles, ECG and eye tracking to predict the user workload and other metrics of assessing readiness of warfighters within a time window of seconds.

2. MOTIVATION FROM THE STATEMENT OF WORK

"In the next phase of IWBP the modeling software will be extended to meet requirements that the end-users have specified that will improve the capabilities and use of the software to support military operations. It will also begin to extend the modeling to other disease states including chronic diseases of concern to the DoD. The work on traumatic brain injury described in this phase of the IWBP is the first example of such studies. "

This provides the motivation for the extension of the Gryphon platform to simulating mild Traumatic Brain Injury (mTBI). In a related vein, Quantum Leap modeling technologies were applied to modeling cognitive readiness of warfighters. The other topics in this summary relate to the broader themes of the IWBP in terms of enhancing situational awareness in complex, dynamic environments and improving operational planning for complex scenarios.

3. BACKGROUND

Quantum Leap Innovations is a technology company focused on the research and development of advanced analytics solutions. We are a leader in the emerging area of Pattern Based Analytics. The Quantum Leap Pattern Based Analytics product family includes: Pattern Based Discovery, Pattern Based Prediction, and Pattern Based Reasoning. Application areas include homeland security, intelligence and Defense, manufacturing, healthcare and life sciences

4. CONTRACT ACTIVITIES

4.1 Extension of Gryphon platform to mTBI

4.1.1 Background:

Under Projects 2 and 3 of the SOW for ONR Contract N00014-09-C-0033, QLI was tasked with development of pilot technology for analysis of medical databases as basis for expanding the Gryphon user community. This expansion of QLI technologies to chronic diseases of concern to the military medical community is initially focused on Traumatic Brain Injury (TBI). The first task (TBI-1) within Project 3 focuses on model building. QLI is collaborating with Dr. Carey Balaban and his team at the University of Pittsburgh to incorporate his extensive behavioral models into QLI technologies.
The Quantum Leap & University of Pittsburgh Traumatic Brain Injury Modeling & Simulation Workshop was hosted by QLI at the Christiana Hilton on July 23, 2009. The collaborating team members include Dr. Carey Balaban, Dr. Dennis K. McBride, Kristofer Younger, Dr. Ganesh Vaidyanathan, Dr. Bin Yu, Dr. Jijun Wang, and Dr. Dawn Defenbaugh. The six-hour brainstorming session resulted in the decision to begin by modeling a well-studied sub-system of the brain, the vestibulo-parabrachial network, and characterizing its role in the co-morbidity of space and motion discomfort (SMD), migraine, and balance disorders commonly associated with mild TBI (mTBI).

An interesting characteristic of mTBI is the degree to which there are individual differences in its production, manifestation, progression, and recovery course. Based on the scientific literature, Balaban has very recently produced a comprehensive model [1] of the neurophysiological and the associated psychological symptoms associated with mTBI (see Quad Chart). This model has been favorably peer-reviewed and was presented on invitation at a recent Samueli Foundation conference. The pre-mTBI VOR components of the comprehensive model have been rendered computationally to simulate the VOR component phenomenology. Classically, loss of balance control and/or dizziness are the most frequent and earliest symptoms associated with mTBI [3] [4], however, VOR testing has been restricted to the slow component of the eye movement. The Quantum Leap Gryphon simulation expresses the magnitudes, directions and timing of both slow and fast phases for normal (pre-TBI) subjects [2]. The clock for the duration of the slow phases reflects brain stem and vestibulocerebellar function. The triggering and targeting of the fast phase reflects both cognitive (higher level) and cerebellar vermis function. The tasks under this project as well as the progress made under each are summarized below:

4.1.2 Task 1: Software architecture design of Gryphon Simulation Environment for TBI

During the current reporting period, the existing Gryphon software architecture for infectious diseases was reviewed for generality and changes were made to the software architecture to support signal processing and individual agent behaviors in modeling vestibular information processing. These changes allowed the team to execute Task 2.

4.1.3 Task 2: Modeling and Implementing the Vestibulo-parabrachial network

After the requisite changes to system architecture were implemented, the next step was to integrate the Balaban and Ariel optokinetic nystagmus model (C.Balaban, M. Ariel, " A beat-to-beat interval generator for optokinetic nystagmus", Biol.Cybern. 66, 203-216, 1992) into Gryphon. This model is an iterative model that attempts to explain slow phase durations occurring during optokinetic nystagmus in terms of an underlying neural clock, and also introduce novel mechanisms for driving "fast phase" eye movements. The neural clock, or basic interval generator, is modeled from an "integrate to fire" neuron model. During the current reporting period, the Balaban-Ariel model was successfully integrated into Gryphon (and is detailed below).

4.1.4 Task 3: Implementation of user interface (UI) for system/parameter adjustment

Figure 1 depicts components of the VOR. The inner ear, visual processing, activity in several areas of the brain (rostral medulla, pons, midbain), and the muscles of the eye all work together to produce target pursuit eye movements. Mild traumatic brain injury (mTBI) disrupts the normal pattern of visual pursuit, and it is the detection and analysis of these disrupted eye movement patterns that are hypothesized to indicate mTBI.

In short, the model of information flow from Balaban and Ariel was parameterized and used to construct a real time simulation of eye movement patterns. In order to examine the validity of the simulation, inputs to the simulation were provided to the simulation in a controlled methodology so as to mimic forced inputs to the rotation of the head. Step, cosine and square wave inputs thus produced the VOR driven eye movements that are portrayed in Figures 3-5. Inspection of the outputs provided initial validity for the model and its simulation.

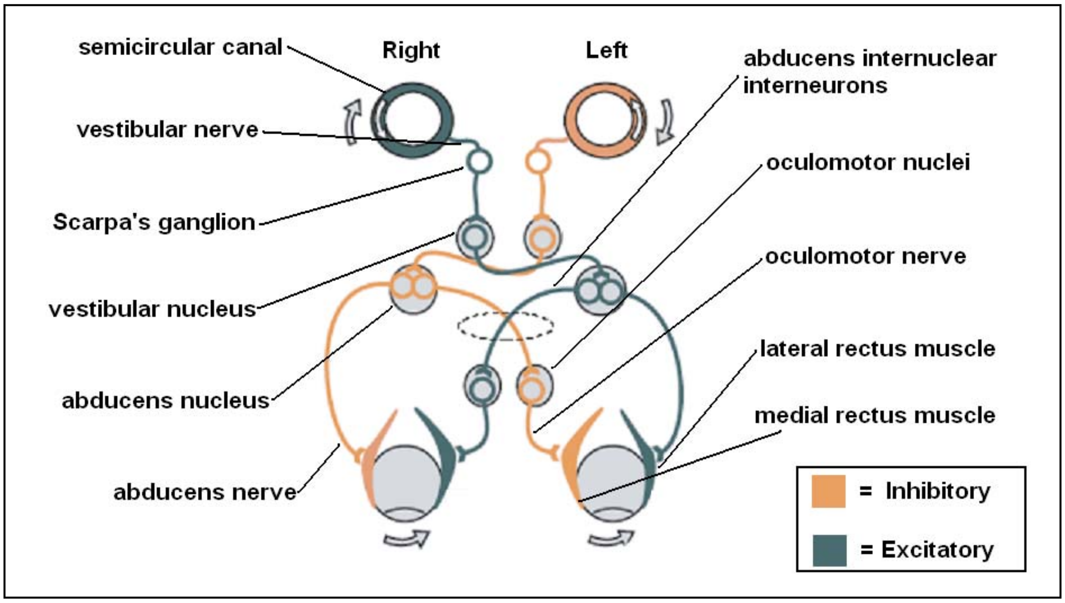


Figure 1: Components of the vestibulo-ocular reflex (VOR)
*(*http://commons.wikimedia.org/wiki/File:Vestibulo-ocular_reflex.PNG *accessed 1/5/2010)*

Figure 2 shows the main screenshot for Gryphon-TBI that implements the Balaban-Ariel nystagmus model. The software was implemented in a modular fashion following a scientific work flow paradigm. Each window provides dynamical traces of the eye position and velocity signals versus time to provide the user with key information across the modeling and simulation process.

Model Structure & User Interface

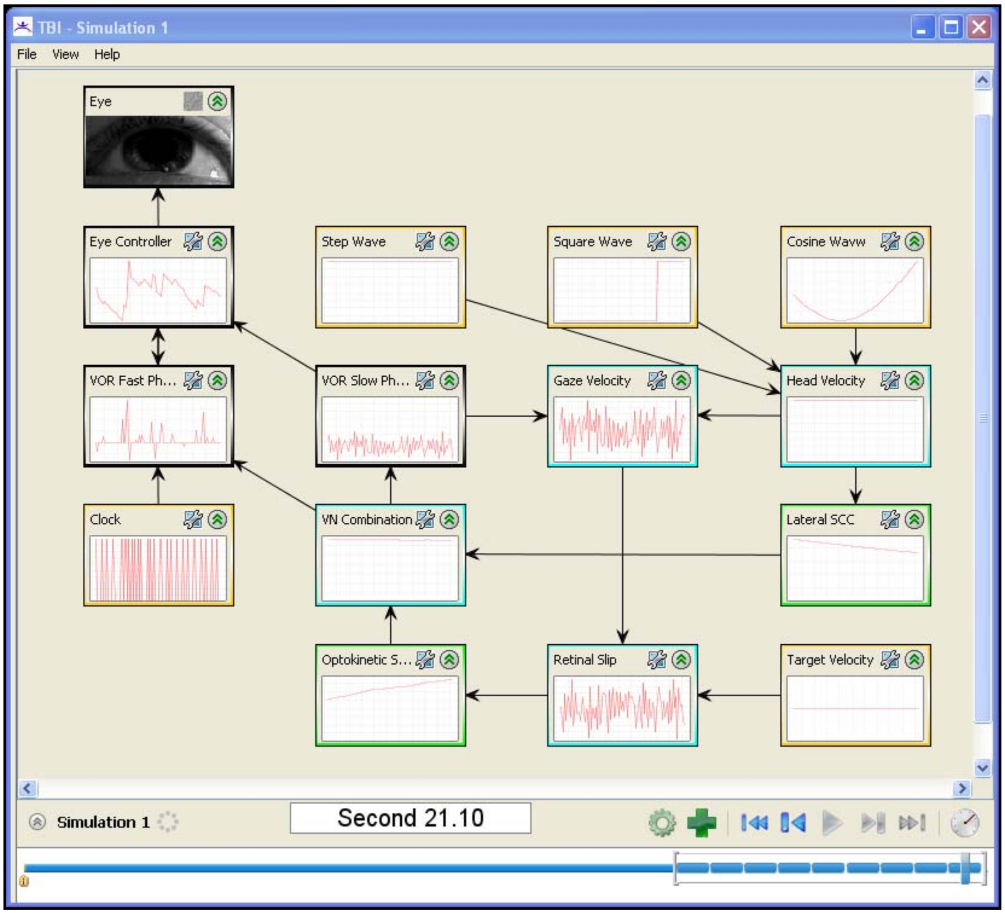


Figure 2: Main screenshot for Gryphon-TBI

4.1.5 Task 4: Evaluation of Vestibulo-Parabrachial network within Gryphon

Figures 3-5 show the response of Gryphon-TBI to different input stimuli. In our initial experiments, we used step, cosine and square wave inputs to drive the optokinetic response. The lower left plot in each of the figures shows how the eye velocity compensates for head velocity in order to present a stationary image to the brain. The residual difference signal, or “retinal slip,” is seen to null out close to zero due to this compensatory effect.

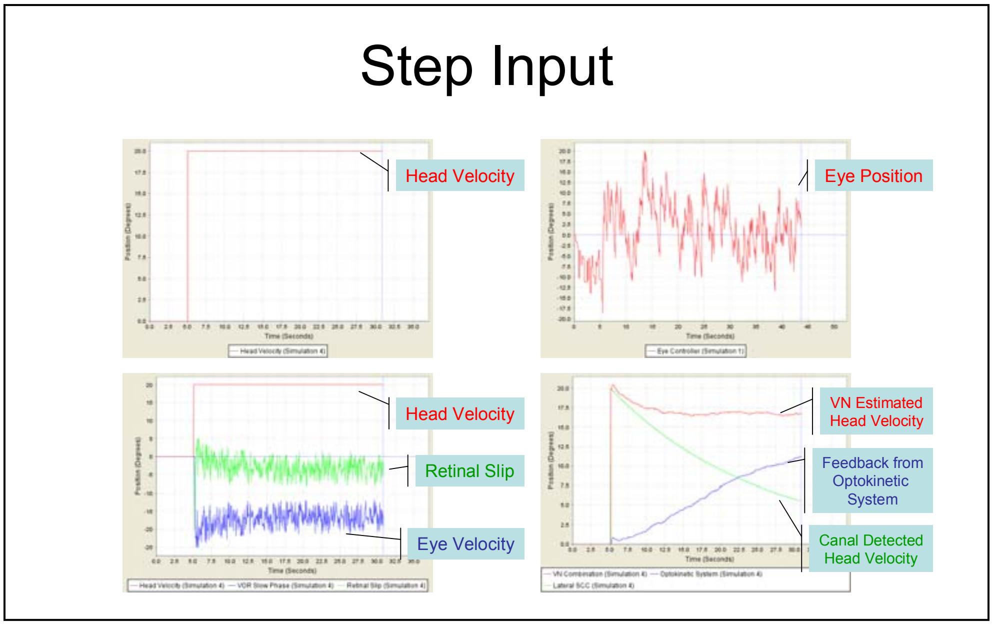


Figure 3: Response of Gryphon-TBI to step input

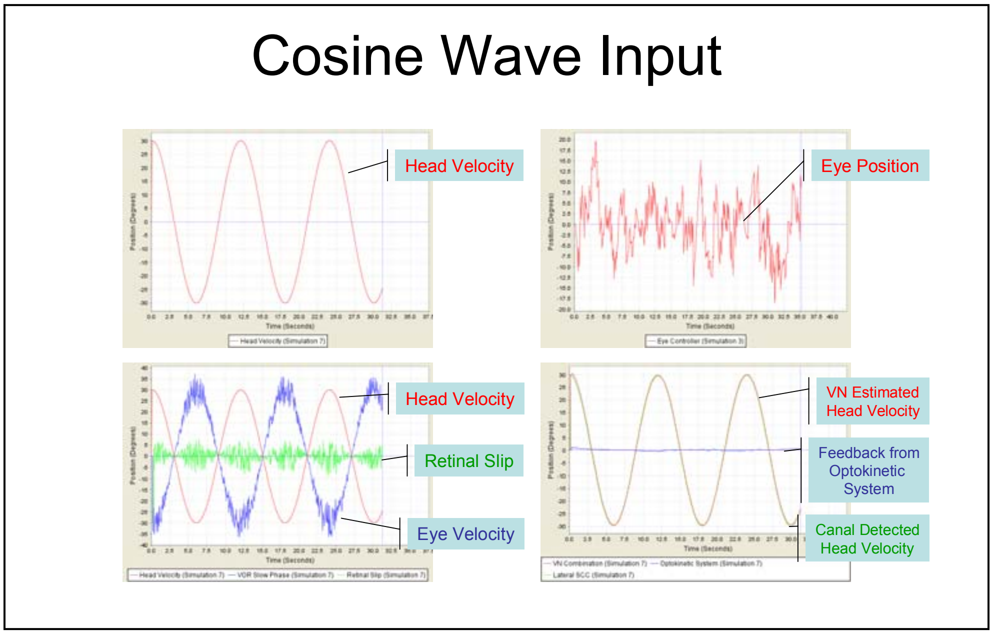
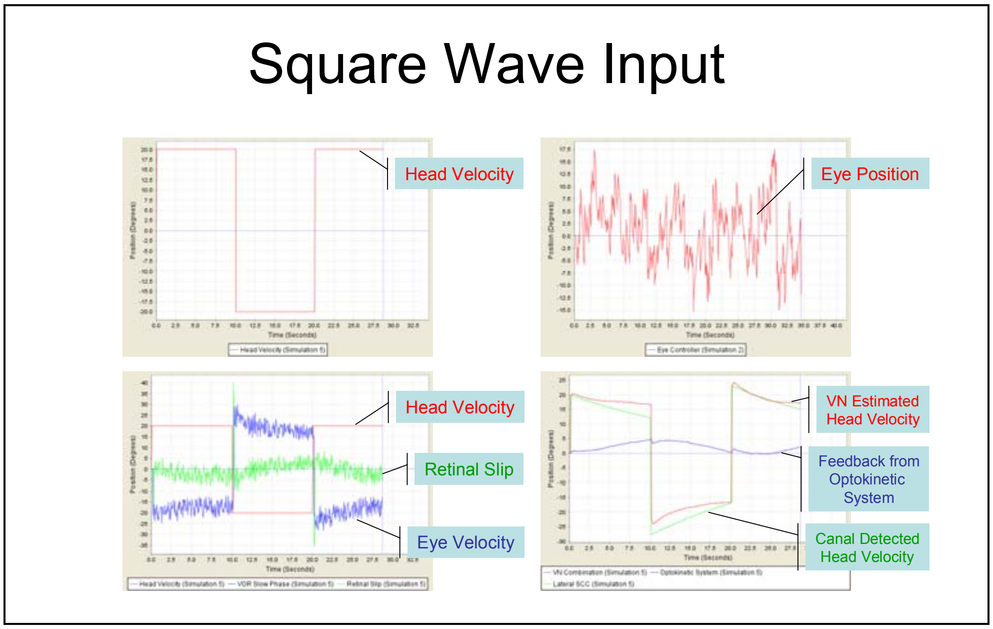


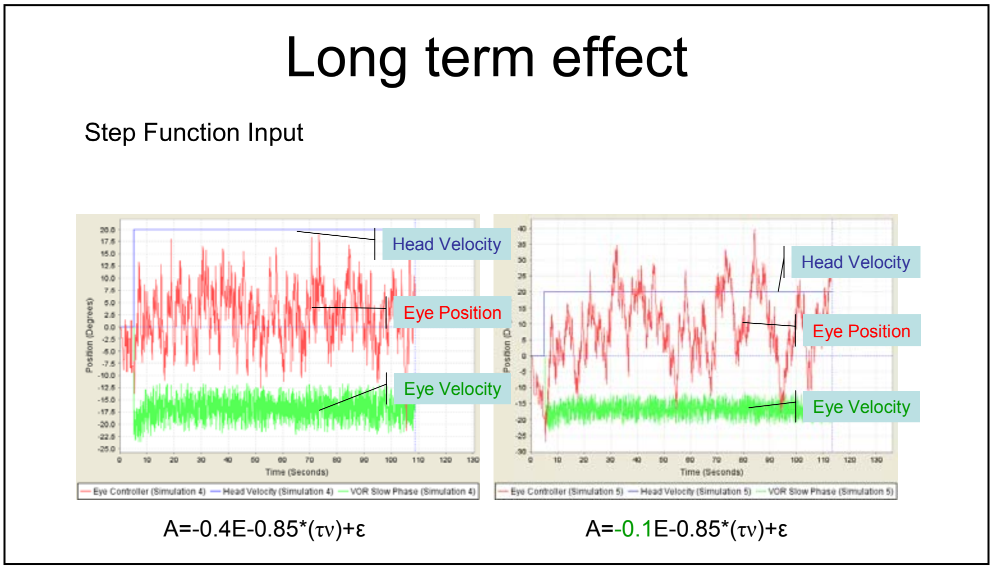
Figure 4: Response of Gryphon-TBI to cosine wave input



Square Wave Input

Figure 5: Response of Gryphon-TBI to square wave input

Figures 6-9 show the long term response of head velocity, eye position and eye velocity to the different inputs parameter values that drive the "fast phase" amplitudes within the Balaban-Ariel model. Note that the right plot in each figure has a lower amplitude coefficient (-0.1 versus -0.4) than the left plot. This represents a weaker compensatory feedback provided by the fast phase, resulting in a more variable trace on the right for eye position. Figure 9 shows similar results for the three input stimuli under slightly different parametric conditions. This type of parametric analysis can provide insight into the relative impact of different model parameters to optokinetic nystagmus.



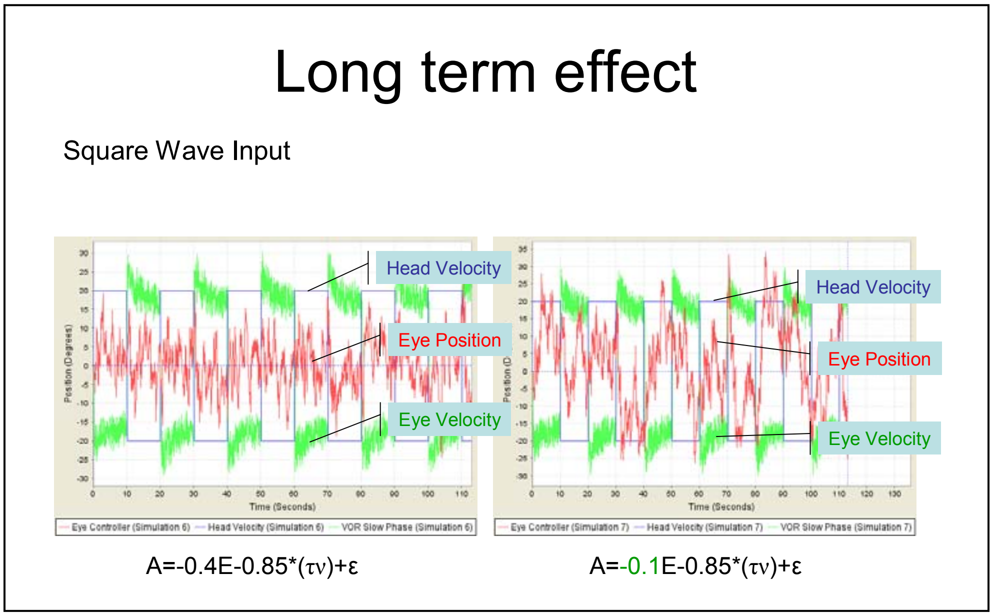
Long term effect

Step Function Input

A=-0.4E-0.85*(τv)+ε

A=-0.1E-0.85*(τv)+ε

Figure 6: Long term effect of "fast phase" amplitude for step function input



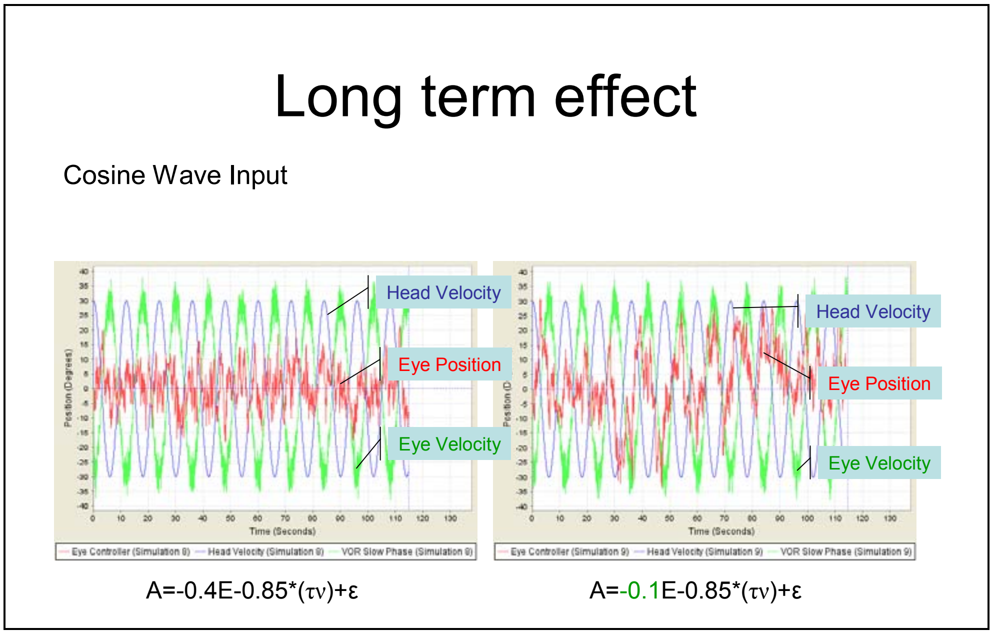
Long term effect

Square Wave Input

A=-0.4E-0.85*(τν)+ε

A=-0.1E-0.85*(τν)+ε

Figure 7: Long term effect of "fast phase" amplitude for square wave input



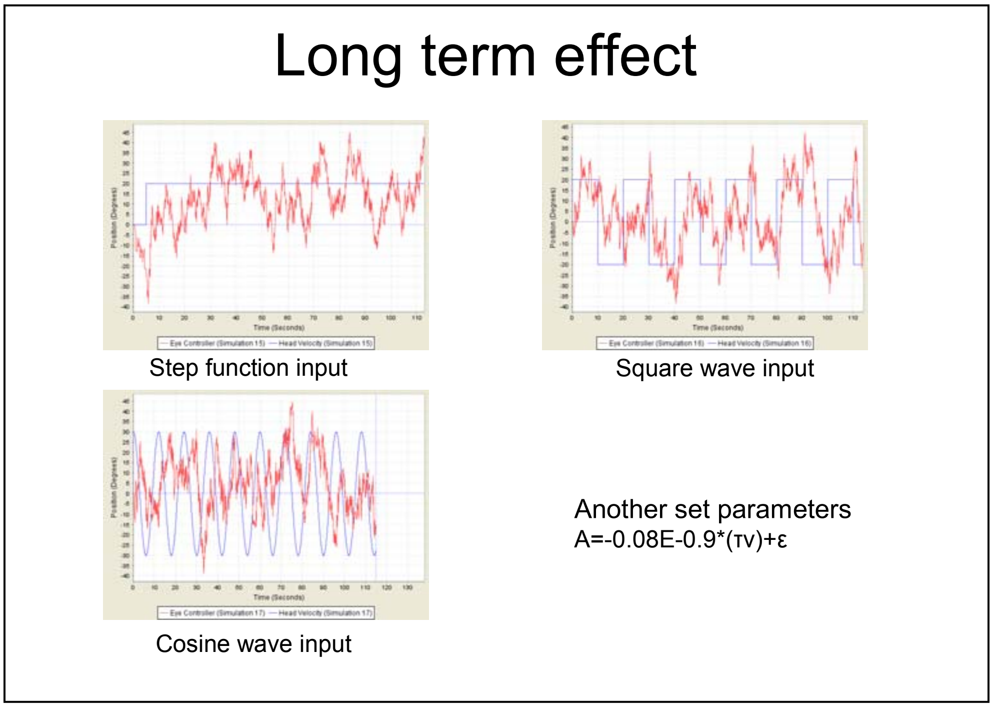
Long term effect

Cosine Wave Input

A=-0.4E-0.85*(τν)+ε

A=-0.1E-0.85*(τν)+ε

Figure 8: Long term effect of "fast phase" amplitude for cosine wave input



Long term effect

Step function input

Square wave input

Cosine wave input

Another set parameters
A=-0.08E-0.9*(τv)+ε

Figure 9. Long term effects for different inputs under new parameter conditions

4.2 Data Analytics – Continued Development of LeapWorks Analytics Platform:

4.2.1 Summary:

Under Project 1, Task GDS-2.2 of the SOW for ONR Contract N00014-09-C-0033, QLI was tasked with development of software to develop component models. This task provided a rationale for the continued development of the LeapWorks Data Analysis platform.

Key defining characteristics of this platform include:

1. The capability of efficiently identifying data relevant to a defined objective or goal within a large, complex data environment.
2. The capability of automatically building predictive models directly from the data.

For example, in the Medical Modeling for Cognitive Readiness project summarized in this report, sensor data streams need to be analyzed from the perspective of cognitive readiness. In general, with the explosive growth of data, the ability to identify the key subsets of data that encode informative patterns against a desired objective can become a key advantage in limited bandwidth and data storage environments. Furthermore, the ability to use these patterns to automatically build predictive models can enable a prognostic capability to situational awareness as a critical component of the IWBP.

The efforts in developing a comprehensive approach to scalable data analysis have straddled both Contracts N00014-08-0036 and N00014-09-0033. Under N00014-09-0033, the efforts have focused on continued validation of the platform. Key aspects of this validation included:

- A comparative study of LeapWorks Predictive Analytics (PA) versus several machine learning techniques implemented in the Weka Open Source Machine Learning Repository. The study was performed across several datasets that were chosen both as representatives of important classes of problems as well as for the different types of challenges that they presented.
- In addition to "quality of results" metrics, the study also compared the performance times across the various methods. Model building time was measured for each method across the datasets as a measure of scalability. Methods were assessed in a two dimensional "quality of result versus model building time" space to assess the resulting tradeoffs.

In addition, extensions to the platform to handle unsupervised pattern discovery- eg. when there is no pre-determined target attribute were formulated. In many environments where it is important to gain situational awareness, an explicit target attribute may be difficult to define up front. In such cases, it will still be important to identify anomalous data behavior as a possible precursor to an unusual event. The proposed extension of the Leapworks platform to handle such situations is summarized below. We anticipate implementing such a capability into our platform as we move forward.

4.2.2 Conceptual Basis for Unsupervised Pattern Discovery:

We have progressively extended our analytics platform to model binary targets, multi-state targets and continuous targets using the same conceptual framework. This has provided a unified basis for developing an efficient analytics kernel that can be embedded in different data environments. To further extend the scope of the platform, we have adapted our Information Theory based framework to address situations where no a priori target has been defined. In this scenario, the objective is to identify unusual associations that occur within the data. These associations, or patterns, typically involve a small set of features and can be viewed as representing *micro-clusters* within the data. Identifying informative micro-clusters within large, complex data sets is complementary to the identification of larger cluster clouds using traditional clustering methods. The increased specificity of such micro-clusters can provide insight as well as form the basis for effective decision making and control strategies. Example applications for unsupervised pattern discovery include cyber security, where the goal is to identify anomalous associations within network data that may be a precursor to a cyber threat. Other applications include the broader domain of anomalous association discovery, for example in a distributed sensor environment.

There are two broad use cases for unsupervised pattern discovery:

a. Discovery of unusual patterns in static data.
b. Discovery of unusual patterns in dynamic data.

Both of these use cases can be addressed using a different scoring metric to assess unusual data distributions based on the Kullback-Leibler (K-L) divergence metric. From Wikipedia (http://en.wikipedia.org/wiki/Kullback%E2%80%93Leibler_divergence):

For probability distributions P and Q of a discrete random variable their K–L divergence is defined to be

$$D_{\mathrm{KL}}(P\|Q) = \sum_i P(i) \log \frac{P(i)}{Q(i)}.$$

Figure 10. Equation 1. Definition of K-L divergence measure

The K-L divergence measures the difference between a probability distribution P and a reference distribution Q.

a. For the static use case (a), the reference distribution can be theoretically derived from the distribution of data across all patterns within an attribute combination under the assumption of statistical independence of the attributes. The K-L divergence between the *actual* distribution of data within an attribute combination and the reference distribution can be used as the scoring metric to drive a genetic algorithm to evolve the attribute combinations that show a high level of statistical dependency.

b. For the dynamic use case b, the reference distribution can be derived from the distribution of data across all patterns within a combination *over an earlier moving window interval*. The K-L divergence between the actual distribution of data within a combination over the *most recent* moving window interval and the reference distribution can be used as the scoring metric to drive a genetic algorithm to evolve the combinations where the associations between the attributes have shown a significant change over the most recent window interval.

   Note: The window interval size should be a user defined parameter that can be adjusted.

In the foregoing discussion, the discovery of patterns that manifest high K-L divergence can enable the identification and visualization of micro-clusters. We can aggregate several patterns associated with the most significant micro-clusters based on a local K-L divergence threshold. Each pattern level K-L divergence would be calculated as a single term in the summation of Equation 1. Aggregating multiple micro-clusters can be useful to isolate an aggregate data subset within a larger data environment that contains multiple anomalous associations. This data can be used for subsequent exploration and analysis.

For some applications, it may be important to identify patterns that involve attributes that are *not* associated with each other – eg. those attribute combinations where the constituent attributes are independent of each other. A good example is when the attributes represent predictive models that are modeling a specific problem – eg. financial forecasting models for trading. In such a case, discovering combinations of models where the individual models within the combination are uncorrelated with each other can be very useful to build more robust combined models. The lack of correlation can help prevent all the models from deteriorating at the same time. Another example along the same line is combining health risk assessment models where diversity of models may be critical to ensure robustness of prediction.

4.3 Optimizing Multi-Ship, Multi-Mission Operational Planning

4.3.1 Overview

This summary serves to detail a compelling naval planning problem that also serves as a template for other complex operational planning scenarios, and provides a natural fit for QLI Modeling, Simulation and Optimization technologies.

4.3.2 Motivation for Navy Mission Planner

This section details the problem definition of the Navy Mission Planner (NMP) as defined by Silva (2009). The details are direct quote or paraphrases.

Problem

“Maritime component commanders employ naval forces in support of the combatant commander. To support the commander’s goals, staff planners in Maritime Operations Centers assign particular ships to particular missions in particular regions at particular times.” (Silva, 2009, p. xv)

“There are many factors involved in building a fleet schedule. Ships flow in and out of theater. Areas of operations typically cover large geographic areas. Some areas require multiple missions to meet the combatant commander’s force requirements, and some missions require support from multiple units.” (Silva, 2009, p. xv)

Inputs

“The planner’s initial NMP input is the set of days covering a finite planning horizon. The user then inputs the planned operating areas into NMP as regions, each of which is an area of the ocean specified by a latitude and longitude at or near its center.” (Silva, 2009, p. 13)

“The user then defines adjacency arcs, representing unobstructed great-circle navigation routes between pairs of regions. NMP then computes and stores the arc lengths (in nautical miles), the shortest path between all regions in nautical miles using sequences of great circle arcs, and transit days (at 16 knots) required for each such path.” (Silva, 2009, p. 13)

“Mission requirements are specified in a list of missions, each of which has a mission type, drawn from a fixed list of types (e.g., air defense, surface warfare, etc., as defined in Chapter IV), a region, and a set of days for which it is required. In addition to the type, region, and day requirements, the planner defines, for each mission, in each region, on each day, a value for accomplishing that mission, and a set of mission dependencies, which define prerequisite missions that must be accomplished simultaneously with that mission, to enable other ships to complete it.” (Silva, 2009, p. 13)

“The last input set is the set of available ships. The operational planner defines the set of ships by hull number and name, start day, start region, and available concurrent mission capability sets (CMCs). The start day is the first day of the planning horizon during which a ship is able to complete mission tasking. A single CMC set is a vector of accomplishment values, one for each mission type, that indicate the fraction of a particular mission that a ship can accomplish concurrently with other missions in the CMC set. One ship can have multiple CMC sets to choose from, but it can only operate under one CMC on any given day. Values less than one indicate reductions in readiness for various issues, such as maintenance or personnel.” (Silva, 2009, p. 14)

Output

“The output from NMP is a set of employment schedules. Each ship’s employment schedule specifies, for each day in the planning horizon, the region in which the ship operates and the assigned CMC set for that ship on that day in that region. NMP provides employment schedules to maximize the aggregate value of all maritime missions accomplished over the planning horizon (Dugan, 2007).” (Silva, 2009, p. 14)

Limitations and Assumptions

“NMP limits the planning horizon to fifteen day windows due to operational limitations on ship employment schedules. One can model a full campaign by solving for a series of fifteen day windows using a “rolling horizon” approach.

NMP calculates transit time based on a 16-knot speed of advance and rounds fractional transit time to represent whole days. NMP rounds days down when the fractional element is less than eight hours. It rounds up when the fraction is greater than or equal to eight hours. In other words, NMP assumes that a unit may participate in missions if it arrives on station with at least two-thirds of a day remaining.” (Silva, 2009, p. 14)

"NMP builds candidate employment schedules through constrained enumeration as an input to the integer linear program." "Path enumeration in NMP begins by reading the user-defined limits on the number of ship schedules, max schedules and max schedules per ship, and the number of stall days, max stall days per ship. A stall day is a day in which a ship remains in the same region it occupied the previous day. The parameter max schedules is the main limit. When the number of schedules reaches this constraint, the enumeration terminates. Reducing the maximum allowable stall days permits NMP to consider a more diverse set of schedules within the number of maximum schedules. Conversely, increasing maximum stall days reduces diversity, but allows a single ship to stay on one long mission without rotating out." (Silva, 2009, p. 19)

Scenario

- Days = 15
- Regions = 16
- Mission Types = 11
  1. Air Defense (AD)
  2. Theater Ballistic Missile Defense (TBMD)
  3. Antisubmarine Warfare (ASW)
  4. Surface Warfare (SUW)
  5. Strike
  6. Naval Surface Fire Support (NSFS)
  7. Maritime Interception Operations (MIO)
  8. Mine Countermeasures (MCM)
  9. Mine Warfare (Mine)
  10. Intelligence Collection (Intel)
  11. Submarine Intelligence Collection (SubIntel)
- Ships = 18 (not all available for whole time)
- Missions = 80

4.3.3 Our Approach

During the current reporting period, we used the Quantum Leap Adaptive Optimization Engine (AOE) to solve the same problem without relying on Silva's (2009) approach that uses constrained enumeration to generate candidate employment schedules for each ship and uses integer programming to optimize the employment schedules to maximize total value of completed missions. We would use the same inputs to the total problem as NMP except:

- Use individual ship speed based on type rather than assuming 16 knots but make same assumption that a unit may participate in missions if it arrives with at least two-thirds of a day remaining.
- No need for user defined constraints on schedules (max schedules, max schedules per ship, and, max stall days per ship)

The AOE can model this problem in a more straightforward way with the outputs as decision variables. There would be a variable for each ship for each day after the ships start day indicating which region it is in on that day. Additionally, there would be a variable for each ship for each day indicating which CMC set to use on that day (picked from two or three choices for that ship). This is less than 540 variables. There would be one constraint per ship to ensure that the schedule is feasible given travel time needed between regions. The objective would be to maximize the sum of the amount of accomplishment of the mission on each day, each weighted by the user defined mission value. A missions accomplishment on a particular day is calculated by summing all the accomplishment values for that mission type provided by all the ships in that region on that day given their assigned CMC set with a maximum value of one. This is the way that Silva (2009) calculated the objective and allows partial accomplishment of missions.

An alternative for the objective is to require missions to be either covered or not on a particular day by rounding the summation of all the accomplishment values for a mission type provided by all the ships in a region down if less than one. This still allows for several ships that are degraded to work together to accomplish a mission but does not allow for partial accomplishment.

In addition to using the Adaptive Optimization Engine, we also developed and tested a refined genetic algorithm and compared our solutions with Mixed Integer Linear Programming (MILP). Results are summarized in Appendix A. The results show that with the limitation of a relatively small number of candidate schedules being used to drive the MILP method, both the genetic algorithm and the AOE converged to better solutions significantly faster than the MILP approach.

4.3.4 Comparison of Optimization Techniques for Mission Planning

4.3.4.1 Summary of Techniques

In every technique the objective is the total amount of missions completed weighted by the mission values. (mission values are part of the initial user input).

Once every technique is completed the solution is created by taking the assigned schedules for each ship and the ship CMCs for each day.

MILP

The Mixed Integer Linear Programming problem is based off of the thesis by Silva.[1] It contains two parts. First the program generates a random set of schedules for each ship, each schedule lists where a ship is on each day.

Then the MILP is given the random set of schedules, the ship CMCs and the missions, mission values and mission requirements. The MILP then chooses:

- The schedule for each ship
- The CMC for each ship on each day
- The amount of each mission completed on each day. (It must choose this because some missions compete for the same work from the same ship.)

AOE

The AOE problem is given variables to determine:

- For each ship and day where is the ship on that day?
- For each ship what CMC is assigned to it on each day?

At each evaluation the ship schedules are "corrected". If the ship cannot get from one assigned location to another location in 1 day, the ship is assigned a transit day at the necessary point in the schedule and all subsequent location assignments are moved back 1 day. (the last day assignment is dropped.)

At each evaluation if there is competition between missions for work the mission with the higher value is given that work.

[1] Silva, Robert A. (2009). Optimizing Multi-Ship, Multi-Mission Operational Planning For the Joint Force Maritime Component Commander. (MS Thesis in Operations Research, Naval Postgraduate School).

Genetic Algorithm

The GA problem is given variables to determine:

- For each ship what is the complete schedule of the ship? (These are directly determined by the GA)
- For each ship what CMC is assigned to it on each day? (This is determined by a second routine inside the evaluation function of the GA)

Each member (complete genome) is a map between all ships and the ship's schedule.

For the mutate function some assigned days for some assigned ships are randomly changed, and the resulting schedules then corrected in the same way as in the AOE.

For the cross function some assigned days for some assigned ships are swapped, and the resulting schedules then corrected in the same way as in the AOE.

During each evaluation phase the complete set of possible CMC assignments for all ships in a given Region * Day are evaluated and the optimal CMC assignment is cached. (This can be improved in future work).

At each evaluation if there is competition between missions for work the mission with the higher value is given that work.

4.3.4.2 Comparison/Evaluation of Techniques

We perform 3 tests using a 20 minute test of each algorithm using each of 80, 160 and 240 missions. We also perform a test of each algorithm on the 80 mission problem for 12 hours.

(1) MIP

First we consider the MIP against the other two algorithms. As we can see in the 20 minute, 80 mission trials the MIP takes much longer to get an initial solution, and after quite a while gets a solution that is still not very close to the GA or AOE. We also notice that in one of the 5 runs the MIP was unable to get *any* solution in the first 20 minutes.

When we consider the 12 hour test (with 1000 schedules for the MIP) we see that the MIP does achieve a score of ~3500, but that it takes almost an hour and a half to get this solution. Both the AOE and GA get a better solution much sooner.

Although the MIP does have a configurable parameter (the number of schedules), it appears that regardless of how you set this parameter the MIP will never beat the AOE or GA in terms of score vs run time. In other words for any run time the MIP will give a worse score in that amount of time than the GA or AOE. (This may not be true in the case where the MIP considers *every* possible schedule, but this would take a very long time and huge amounts of memory.)

In conclusion the MIP is not the best algorithm to use.

Note: The best solution found in the thesis paper using the MIP is 3503.5, which is just barely better than our best solution of 3502.75. The difference is likely due to random generation of different schedules.

(2) GA & AOE

The GA and AOE are much more competitive. In the 12 hour run we see that the GA quickly finds a peak solution. The GA finds its best solution within the first hour. The AOE takes much longer but slowly and incrementally improves its solution until after 12 hours it has a solution that is almost as good as the GA. (The GA best is 3741.25, the AOE best is 3727.15 [due to programming mistake the AOE is an average of 5 runs]).

For the 80 mission and 160 mission runs we see that the GA always dominate the AOE, though the AOE comes close to the GA at about 1 minute before diverging again. After a short period of time the GA levels off. The AOE continues to make small improvements though never reaching the GA's score.

For the 240 mission solution we see that the GA starts out better than the AOE and ends better than the AOE, but that there is a period of time (between 1 and 4 minutes) where the AOE is doing better. In fact the GA has this weird shape where it makes small improvements for a while before quickly making a large improvement and then leveling off. This is likely due to the design of the GA, which I describe later.

In almost all cases the GA does better than the AOE, though given enough time the AOE comes very close to the GA. The one area where the AOE may do better than the GA is in scenarios where the number of missions and ships is very high relative to the number of regions; I discuss this more when describing the GA algorithm.

(3) GA Algorithm

The GA algorithm has 2 parts, it uses a typical GA to create a pool of schedules and then cross/ mutate those schedules in a search for a better schedule. When evaluating each schedule the program must determine the best CMC assignment for each set of ships at a particular day / region. In this draft of the GA we naively check all combinations and choose the optimal combination, caching the score. Thus on the first iterations of the GA there is a lot of work to generate this complete map of (ship set) * region *day -> best score. In later iterations the most of the scores are cached and so the GA runs much more quickly.

This exhaustive search can potentially be dramatically improved with the enhancements listed below:

1) Use a separate GA on each of these subset problems, run the GA for only a very short period of time on each eval, and cache the best score. Based on how far the score is from theoretical optimum, resolve the sub GA each time we get the cached score. This way we will use less time in the early computations, and still get an optimal solution in the later computations.
    a. Rather than using an expensive GA we could also try using a simple random search on only some subsets early in the problem.
2) Use a MIP to solve this smaller problem. Because this problem is much smaller (all we need to do is choose the optimal set of CMC on a single day in a single region) the MIP should be very fast for this problem.
3) Consider other techniques for caching the solutions. For example if we know that ships (A,B,C) provide an optimal solution, then we know that ships (A,B,C,D) also provide an optimal solution, and we don't need to resolve for (A,B,C,D). Is there some way to easily store/retrieve this relationship?
4) There may be other techniques for solving this sub problem, that we can research and focus on.

Ultimately the big question is: How frequently does the area of overlap between the AOE and GA occur? In what particular dimensions of the problem does it occur? How do these different suggestions (above) affect it? How large a discrpency is there between the AOE and GA during the overlap?

Further research and study on these questions could be useful in improving the GA (and potentially the AOE too).

4.3.5 Results of Comparison

A. 80 missions

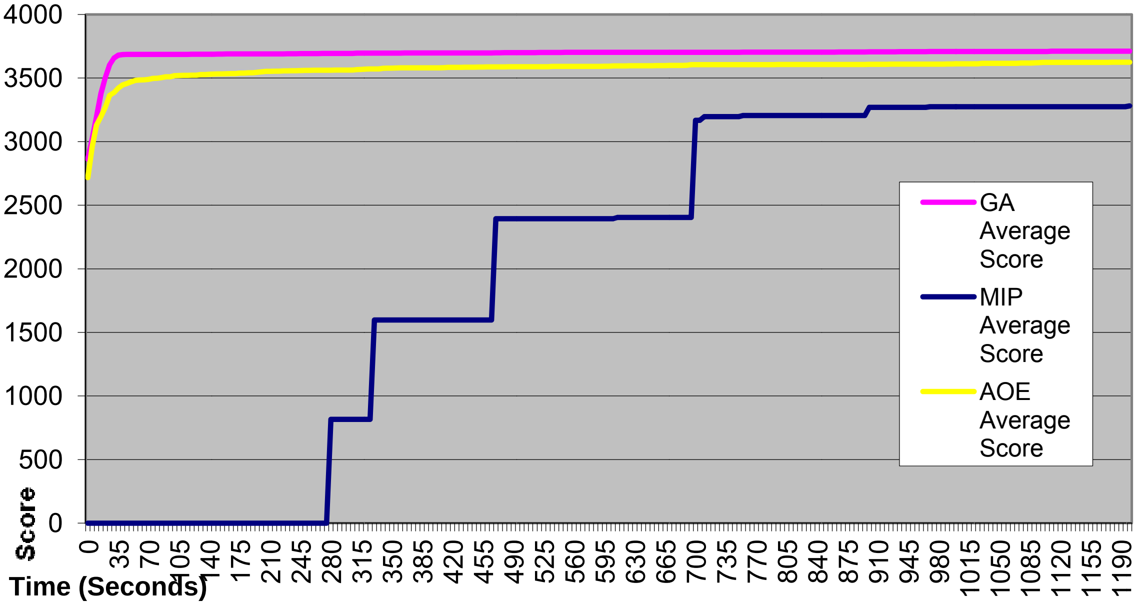


Figure 11: Comparison of techniques for scheduling 80 missions. Plotted is the overall score for the solution versus time.

B. 160 missions

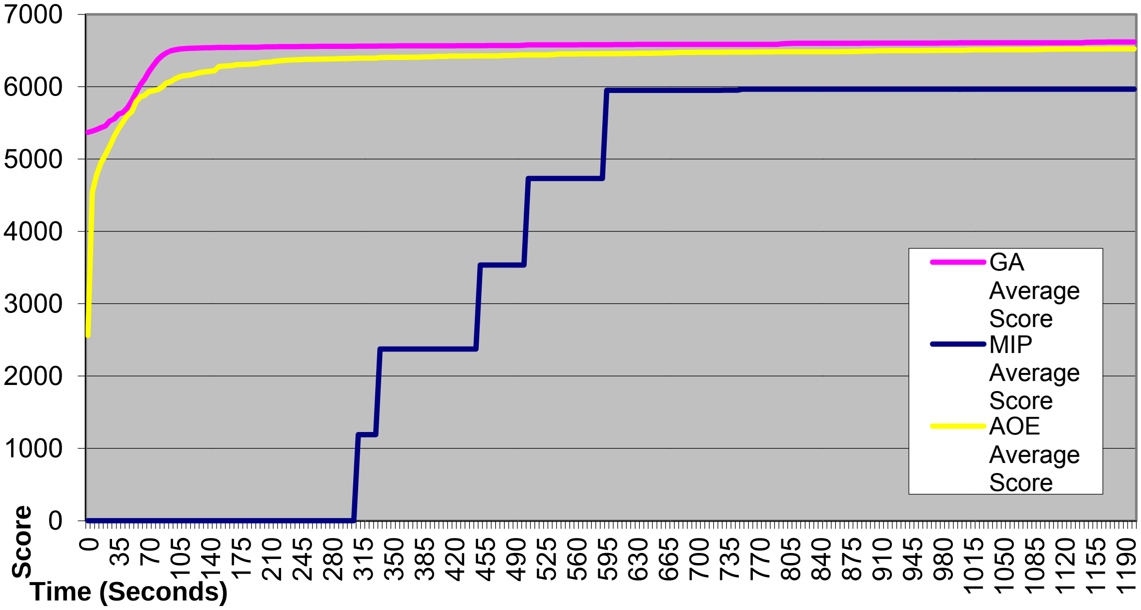


Figure 12: Comparison of techniques for scheduling 160 missions. Plotted is the overall score for the solution versus time.

C. 240 missions

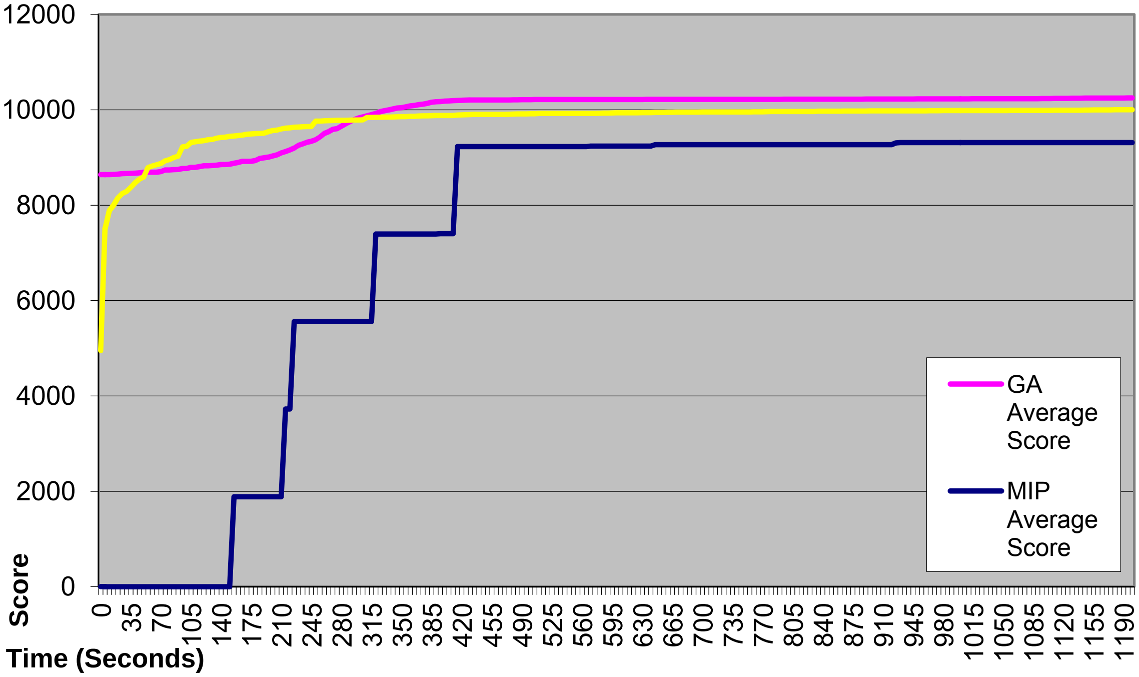


Figure 13: Comparison of techniques for scheduling 240 missions. Plotted is the overall score for the solution versus time.

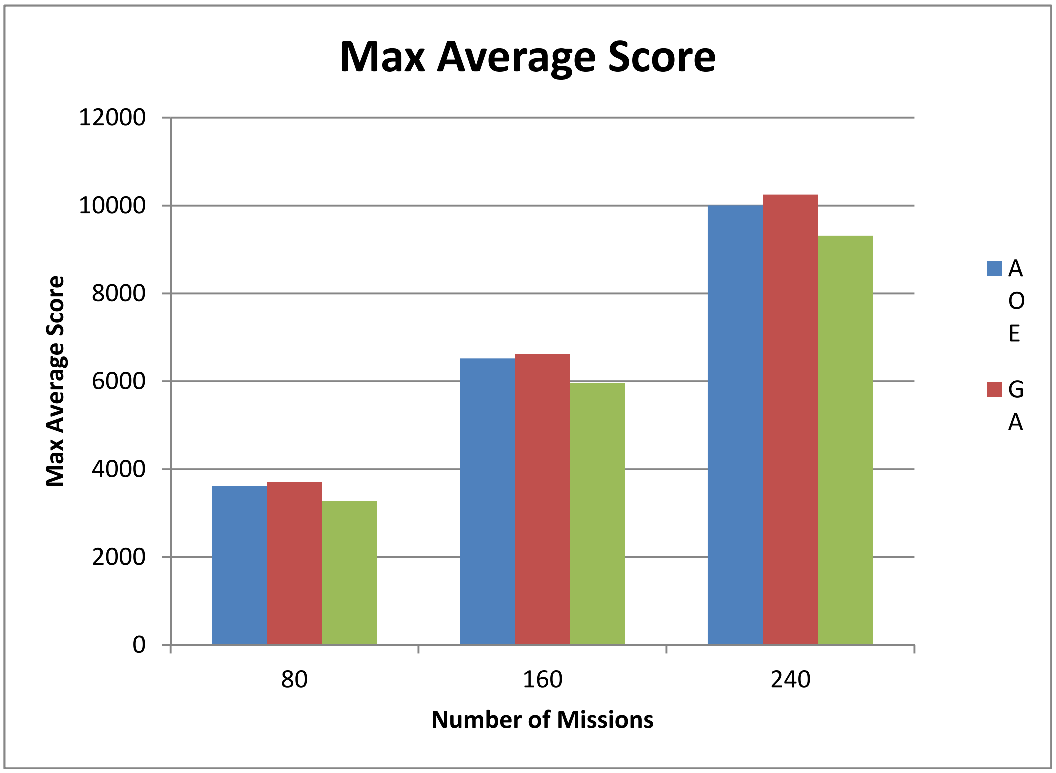


Figure 14: Summary of Performance across the three techniques

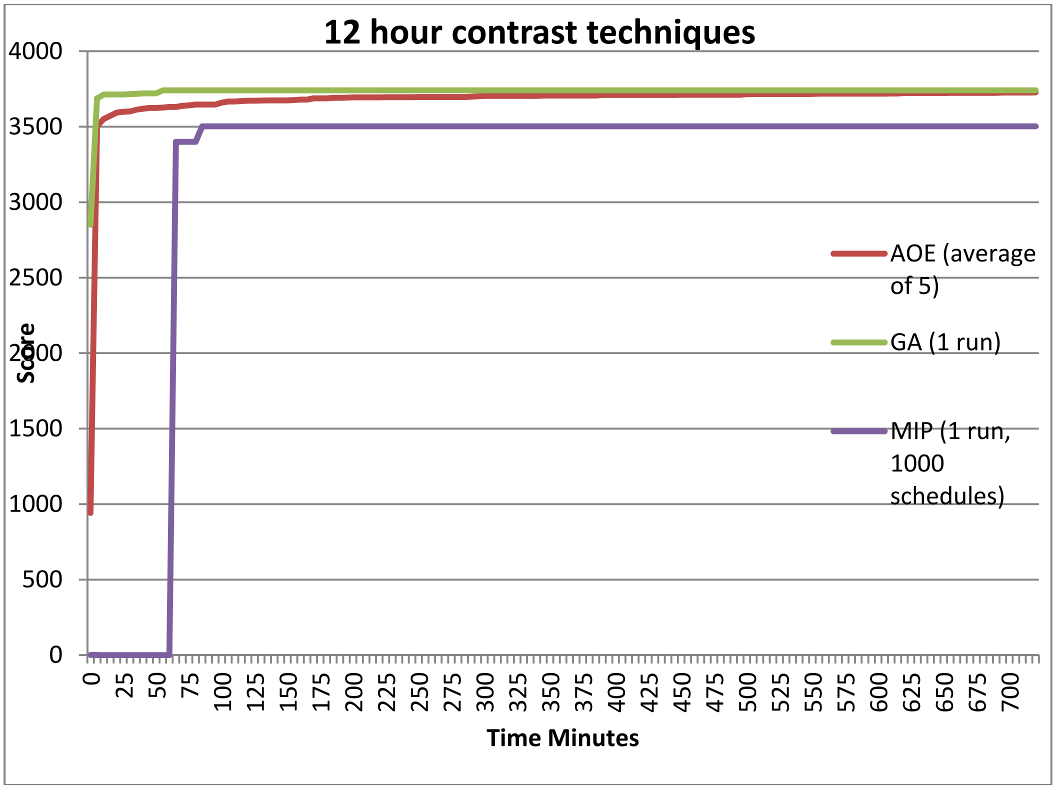


Figure 15: Long time convergence studies of three techniques

4.4 Medical Modeling for Cognitive Readiness

4.4.1 Motivation

Neurophysiological sensors such as ECG, EEG and eye tracker have been extensively investigated as a means of assessing cognitive readiness of warfighters. Many metrics and techniques have been addressed within the DARPA AugCog program and DoD programs. For example, Berka *et al.* at Advanced Braining Monitoring (ABM) has developed a mental workload metric based on an individual's EEG signal that tracks task demand in mental arithmetic and digit span tasks.[2] The mental workload metric has shown a significant correlation with subjective measures of workload and task performance. Other researchers have focused on eye tracking metrics and found changes in pupil diameter, fixation duration, and blink frequency to be predictive of levels of cognitive demand in a task.[3]

The goal of this project is to develop a reliable model for cognitive readiness assessment using multiple modalities of sensors and adapt the model from groups to individuals in specific tasking environments.[4] Adaptation of a model to different individuals and tasks is one of the requirements in the DARPA AugCog program. Instead of directly using ABM's workload metrics, which have

[2] Berka et al, EEG Correlates of Task Engagement and Mental Workload in Vigilance, Learning, and Memory Tasks, *Aviation Space and Environmental Medicine*, 2007

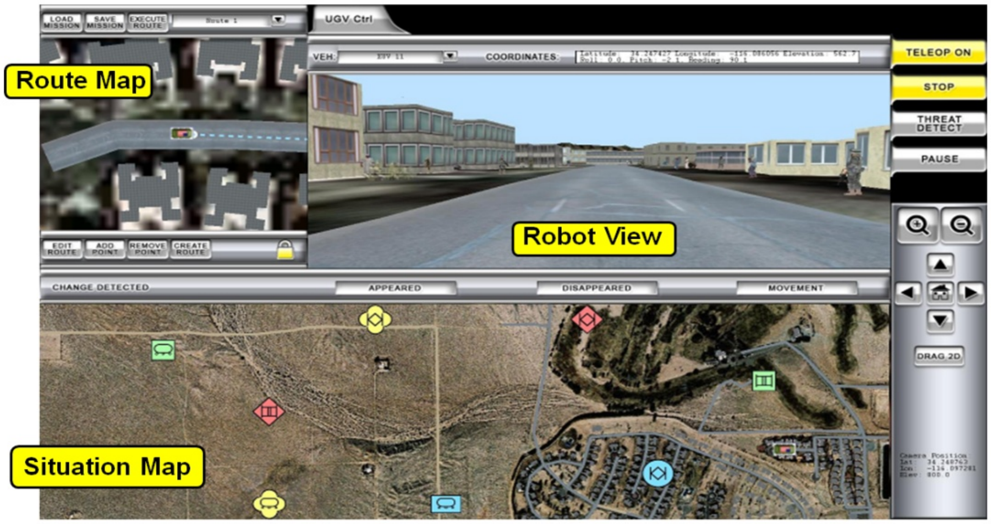
[3] Tsai et al, Task Performance and Eye Activity: Predicting Behavior Related to Cognitive Workload, *Aviation Space and Environmental Medicine*, 2007

[4] Leah M. Reeves, Dylan D. Schmorrow and Kay M. Stanney, Augmented Cognition and Cognitive State Assessment Technology – Near-Term, Mid-Term, and Long-Term Research Objectives, Foundations of Augmented Cognition, 2007

shown no significant changes over task difficulty levels in UAV control, we have developed a suite of data fusion and analysis technologies from data cleaning and synchronization for multiple sensor data streams, feature extraction, to model development and testing. We combined subjective ratings, user profiles, ECG and eye tracking to predict the user workload and other metrics of assessing readiness of warfighters within a time window of seconds. EEG data was dropped from the study due to limited participants.

4.4.2 Mixed Initiative Experimental (MIX) Testbed

The UCF team has worked with its DoD partners to construct a testbed that leverages the benefits of the MATB (Multi-Attribute Task Battery) and VBS2 and developed a mixed initiative experimental (MIX) testbed.[5] The MIX Testbed is a moderately high fidelity simulated environment designed to test human-robot interactions in various automated conditions, such as the formation of mixed-initiative teams and the automated allocation of tasks and resources among robots and operators. The MIX testbed includes the capability of assessing performance in a multi-task environment and the detailed logging capabilities that permit simultaneously synchronizing the data from multiple physiological sensors with performance events. To support data analysis and event synchronization, every data record that is added to a log file includes a Universal Coordinated Time (UTC) and a Simulation time value.
Figure 16 shows the MIX testbed operator interface consisting of a route map (top left) used for direction reference when the vehicle is in a manual control mode or a waypoint reference when the vehicle is in static control mode. The streaming video feed of vehicle way (top right) was used for guiding the operator to travel through the terrain. The bottom half of Figure 16 employed a change detection task in which operators were asked to detect changes in icons (e.g., appearance, disappearance, and move).



Route 1

VEH: UGV 11

Latitude: 34.247427 Longitude: -116.086056 Elevation: 562.7
Roll: 0.0, Pitch: -2.1, Heading: 90.1

Camera Position
Lat: 34.248763
Lon: -116.097281
Elev: 800.0

Figure 16: User Interface of the Mixed Initiative Experimental Testbed (MIX)

4.4.3 Experiments

The multi-tasking MIX environment consisted of a three condition control task in which an operator was required to navigate a ground vehicle through a route.

[5] Daniel Barber, Sergey Leontyev, Bo Sun, Larry Davis, Jessie Y.C. Chen, and Denise Nicholson, The Mixed-Initiative Experimental Testbed for Collaborative Human Robot Interactions, . International Symposium on Collaborative Technologies and Systems, 2008

- In the manual or teleoperation control condition, operators steered the vehicle with a joystick through pre-designed routes.
- In the automated or static condition, the vehicle was directed through routes via waypoints.
- The adaptive automation condition consisted of beginning the scenario in manual and ending the scenario in static or vice versa.

The experiments utilized the MIX testbed to create a multi-tasking environment upon which to access an operator's physiological responses of controlling an unmanned ground vehicle throughout six 9-minute scenarios. Figure 17 describes the design of the experiments for a mixed-initiative human-robot team. Low task load conditions contained roughly 8 icons and high task load altered about 24 icons. An increase or decrease in task load happened in each of the six scenarios in which the task load change implemented at 4 min and 30 sec out of each of the 9 min scenarios.

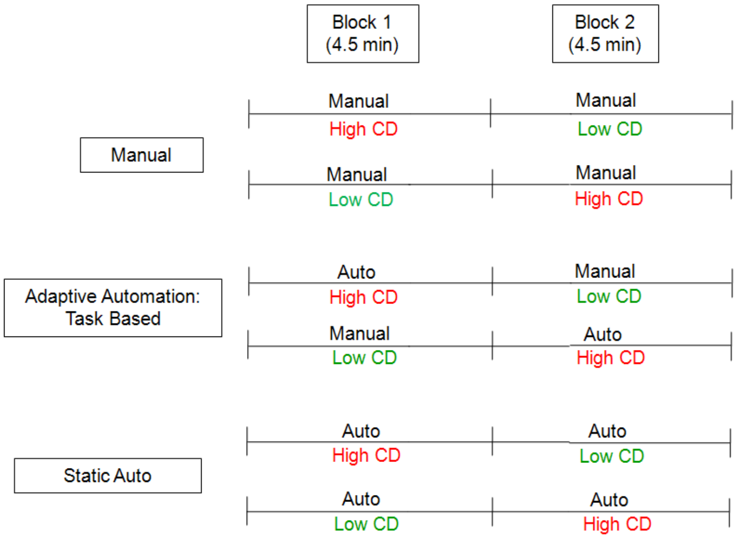


Figure17: Experimental design for control (auto=automation), block, and task load (CD=change detection)

Change detection task load was manipulated by increasing or decreasing the number of icons present in the display. As shown in Figure17, low task load conditions contained roughly 8 icons and high task load altered about 24 icons. An increase or decrease in task load happened in each of the six scenarios in which the task load change implemented at 4min and 30sec out of each of the 9min scenarios.

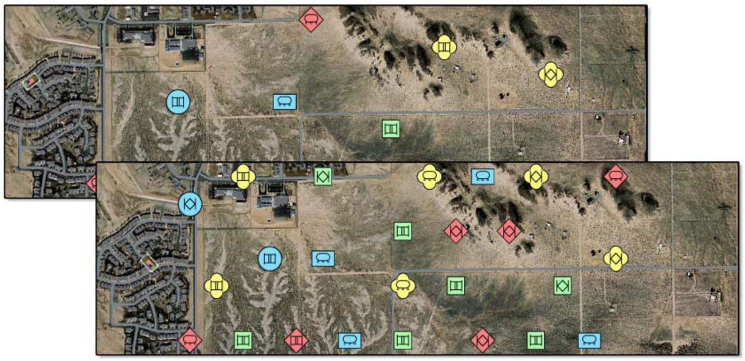


Figure 18: Example of change detection task capabilities.

4.4.4 Experimental Data

Figure 19 shows all physiological sensors in use by the UCF team for data collection. The current sensors consist of Advanced Brain Monitoring (ABM) 6 and 9-channel EEG units, Arrington EyeFrame Eye Tracking Systems, and Thought Technology ECG and GSR systems. Human experiments involving a total of sixty-four college students from the University of Central Florida and the United States Military Academy (25 female and 39 male) have been performed. [6] This data set provided by UCF includes task performance, eye tracking, and electrocardiograph (ECG). Additionally, demographics questionnaires, NASA TLX, and spatial awareness testing data are also provided for each participant.

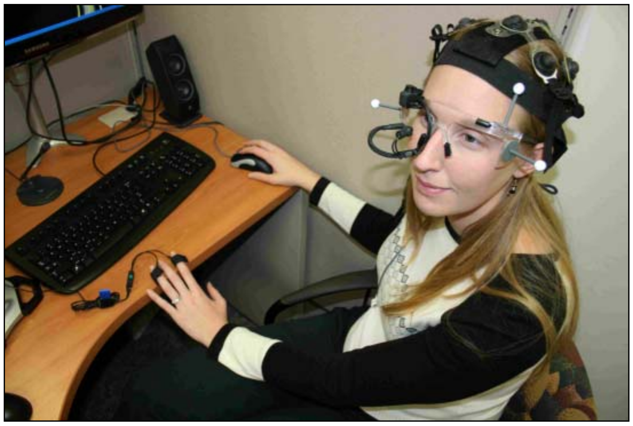


Figure 19. EEG, Eye Tracking, ECG, and GSR connected to a single operator

The details of the experimental data are summarized as follows

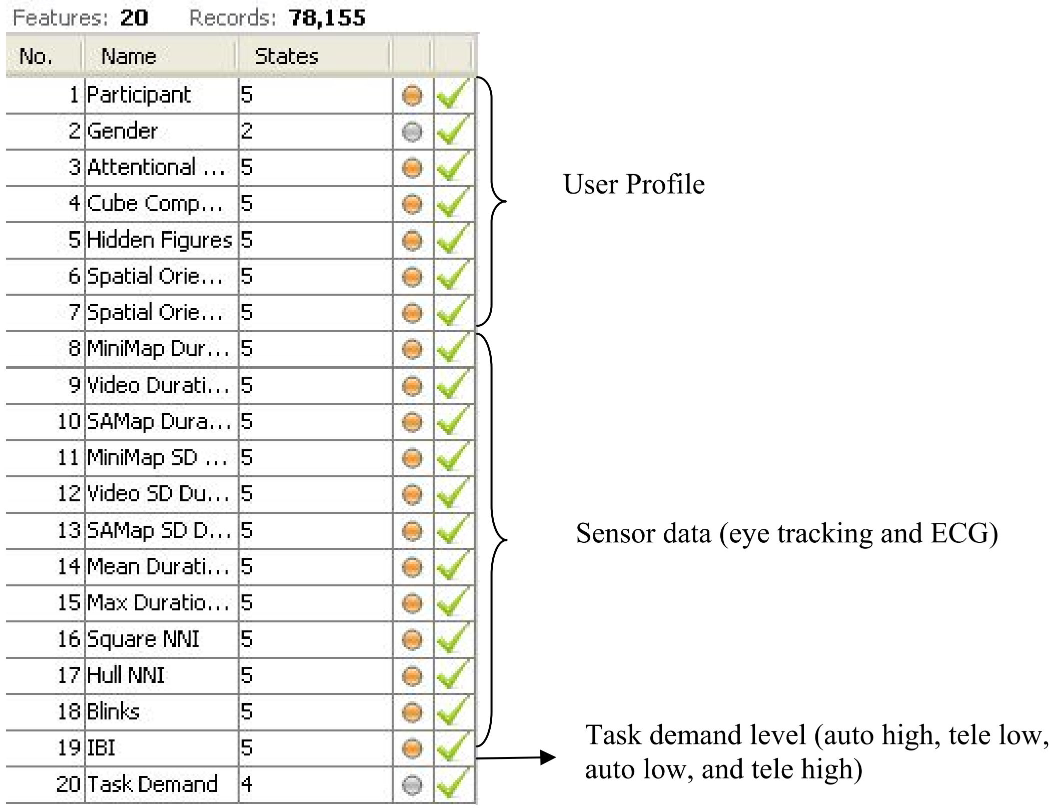
- Task: Four tasks have been employed in the MIX testbed, including threat detection, change detection (appear, disappear, move), situation awareness and audio response (call sign).
- User profile and capability: These include demographical information (e.g., gender and age) and spatial/control capability testing results (e.g., attentional control and cube comparison)
- Sensor data: These include Interbeat Interval (IBI) and Heart Rate Variability (HRV) for ECG data, Nearest Neighbor Index (NNI), Saccades, Blinks, Fixations for eye tracker data.
- System data: These include events and actions for change detection and thread detection, and log files for audio response, vehicle position and orientation.
- User Self Assessment: After each experiment, self report questionnaires NASA-TLX were used to quantify workload. Operators rate their perception of workload for a given task by providing ratings of 0 - 100 with 100 representing the highest perceived level of workload.

4.4.5 Data Preprocessing

There are two steps to generate a synchronized data table for data analysis and exploration in this project: data fusion and pre-processing.

[6] Lauren Reinerman-Jones, Keryl Cosenzo, and Denise Nicholson, Subjective and Objective Measures of Operator Sate in Automated Systems, International Conference on Applied Human Factors and Ergonomics, 2010

- Data Fusion: The sensor data sets are captured at different frequencies and have to be synchronized. In order to fuse the data from different sources, a configuration file in xml was developed to specify 1) time scale and shift, 2) conditions of including whole or part of the data, 3) user profile and capability testing information, 4) sensor and system data necessary to include. Additionally, for each data source we can specify which variable we want to include, their new variable names in the integrated file, and aggregation/de-aggregation functions used to synchronize data within a time block.
- Data Preprocessing: There are two issues after we merge different data sources into a synchronized data table, task demand level generation and user performance data generation. In this study we mapped task states None, Correct, Wrong/Incorrect, Invalid/Noresponse to -1, 0, 1, and 2. We combined UCF task responses into a overall task response using value -1, 0, 1 and 2, where -1 means no task, 0 means a correct response to tasks, 1 means at least one wrong response to a task, and 2 means at least one unnecessary response to a task. We utilized RapidMiner (the most "popular" data mining tool according to KDNuggets 2010[7]) to filter data for a single UCF task. For example, we generated the data set for Change Detecting task only by filtering out any other data that associate with either Threat Detection task or Audio task.



Features: **20** Records: **78,155**

No.	Name	States		
1	Participant	5	●	✓
2	Gender	2	○	✓
3	Attentional ...	5	●	✓
4	Cube Comp...	5	●	✓
5	Hidden Figures	5	●	✓
6	Spatial Orie...	5	●	✓
7	Spatial Orie...	5	●	✓
8	MiniMap Dur...	5	●	✓
9	Video Durati...	5	●	✓
10	SAMap Dura...	5	●	✓
11	MiniMap SD ...	5	●	✓
12	Video SD Du...	5	●	✓
13	SAMap SD D...	5	●	✓
14	Mean Durati...	5	●	✓
15	Max Duratio...	5	●	✓
16	Square NNI	5	●	✓
17	Hull NNI	5	●	✓
18	Blinks	5	●	✓
19	IBI	5	●	✓
20	Task Demand	4	○	✓

User Profile
Sensor data (eye tracking and ECG)
Task demand level (auto high, tele low, auto low, and tele high)

Figure 20: synchronized data table for data analysis and exploration

Figure 20 describes the features of a data table for task demand analysis and exploration after data fusion and pre-processing. The data for each participant include user profile, sensor data and task demand level. In our study, the task demand level was categorized into four levels: auto high, tele low, auto low and tele high.

[7] http://www.kdnuggets.com/polls/2010/data-mining-analytics-tools.html

4.4.6 Model Development for Cognitive Readiness Assessment

We studied the feasibility of developing real-time individualized models from group models for three tasks: task demand, threat detection and change detection. Many techniques have been developed for classifying neurophysiological data for cognitive state assessment. These include linear/non-linear regression, artificial neural networks (ANNs), and decision trees.[8] Due to the lack of experimental data and differences of tasking environment, there is no unanimous conclusion for the performance of these classifiers.
In this project we selected the Naïve Bayes (NB) classifier as the foundation for model development, since a Naïve Bayes classifier can be easily trained for groups and can adapt to individuals. During the adaption process, the weights of some variables are adjusted to reflect the distribution changes of those variables from groups to individuals. There is no easy method to quickly adapt a model in ANNs and decision trees from groups to individuals. Specifically, we adopted a variant of Naïve Bayes classifiers – Aggregating One-Dependence Estimators (AODE). Many techniques have been developed to retain NB's desirable simplicity and efficiency while alleviating the problems of the attributed independence assumption. Previous experiments showed that AODE can obtain the accuracy improvements derived by Lazy Bayesian rules and Super Parent Tree Augmented Naïve Bayes (SP-TAN) without those techniques' high computational overheads. [9]

4.4.7 Task Demand

We first studied the development of group and individual models for task demand.

The data for each participant includes user profile (e.g., gender, attentional control, cube comparison and hidden figures), sensor data and task demand level. In our study, the task demand level was categorized into four levels: auto high, tele low, auto low and tele high. The sensor data features used for task demand include, MiniMap/Video/SAMap Fixations, MiniMap/Video/SAMap Duration (ms), MiniMap/Video/SAMap SD Duration, Fixations, Mean Duration (ms), SD Duration for eye tracking, IBI and SD IBI for heart rates, NNI and blinks for eye tracking.

We first tested the accuracy of groups using AODE, where the model was trained by 70% of the data. The remaining 30% of the data was used for testing. The results showed that we can predict the task demand at the group level with over 70% accuracy.

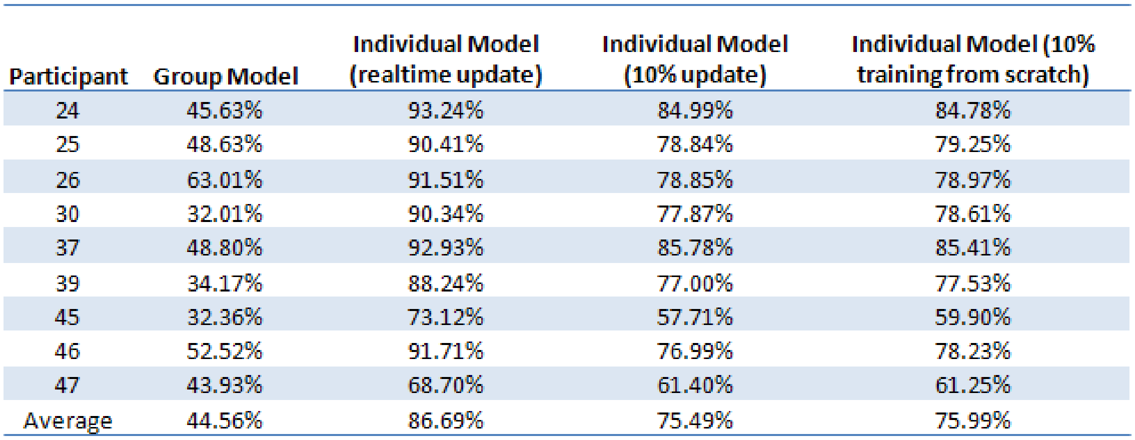
In the second study we used the data from some participants as training data and then we applied the model to predict the task demand level for the rest of the participants. Specifically, we trained the model using the data from all participants except for 24, 25, 26, 30, 37, 39, 45, 46, and 47. And then we tested the model using the data for participants 24, 25, 26, 30, 37, 39, 45, 46, and 47. The accuracy we got is 43.71%. The results indicated that, due to inter-individual differences, it was very hard to directly predict the performance of participants using the model trained by the data from other operators.

[8] Patrick L. Craven, Nadya Belov, and Patrice Tremoulet, Michael Thomas, Chris Berka, Dan Levendowski,
and Gene Davis, Cognitive Workload Gauge Development: Comparison of Real-time Classification Methods, Augmented Cognition: Past, Present and Future. D. Schmorrow, K. Stanney and L. Reeves (eds), 2006

[9] Geoffrey Webb, Janice Boughton and Zhihai Wang, Not so naïve Bayes: Aggregating one-dependence estimators, Machine Learning, 58(1), 5-24, 2005

An alternative to better predict the performance of a participant is to calibrate the group model using the data from the participant before prediction. In the third study for task demand we tried to develop a real-time individualized model in three schemes.

- Individual Model (realtime update), where the group model is constantly updated by the data for a given participant.
- Individual Model (10% update), where the group model is first trained by 10% of the data for a given participant.
- Individual Model (10% training from scratch), where the model is learned from scratch with 10% of data for a given participant.



Participant	Group Model	Individual Model (realtime update)	Individual Model (10% update)	Individual Model (10% training from scratch)
24	45.63%	93.24%	84.99%	84.78%
25	48.63%	90.41%	78.84%	79.25%
26	63.01%	91.51%	78.85%	78.97%
30	32.01%	90.34%	77.87%	78.61%
37	48.80%	92.93%	85.78%	85.41%
39	34.17%	88.24%	77.00%	77.53%
45	32.36%	73.12%	57.71%	59.90%
46	52.52%	91.71%	76.99%	78.23%
47	43.93%	68.70%	61.40%	61.25%
Average	44.56%	86.69%	75.49%	75.99%

Figure 21: Experimental results for task demand

Figure 21 describes the experimental results for task demand using different modeling schemes from groups to individuals. The individual differences vary for different participants, but group models in general cannot capture those individual differences and cannot predict the performance well for a given participant. However, it is possible to calibrate the group model using the data from a participant and use the calibrated model to accurately predict the performance of the given participant. The accuracy ranges from 70% to 90%. There is no significant difference for individual models calibrated from group and individual models learned from scratch.

4.4.8 Change Detection

A change detection task was introduced to model the responses of an operator to changing environments in MIX. Three types of changes occurred – appearance, disappearance, and location movement –an equal number of times at an equal number of two rates, 10 or 15 second randomized, to limit the possibility of participants anticipating a pattern. Change detection task load was manipulated by increasing or decreasing the number of icons present in the MIX display. Low task load conditions contained roughly 8 icons and high task load conditions altered about 24 icons.

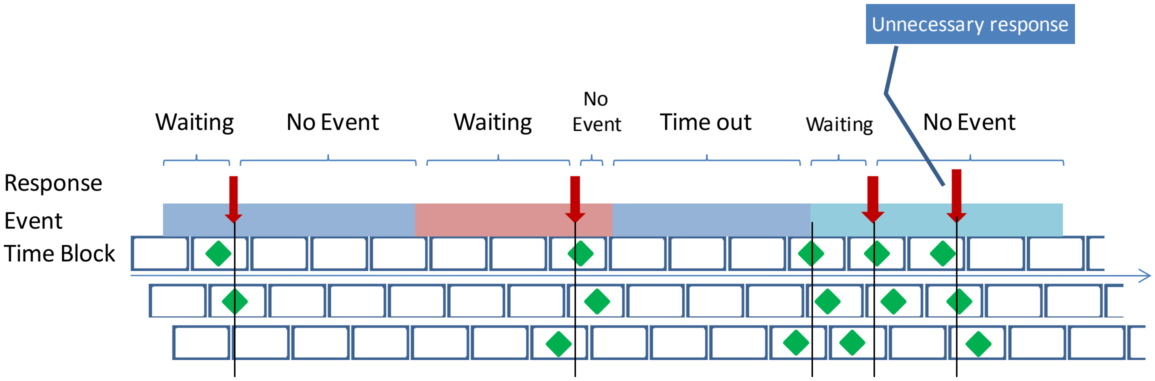


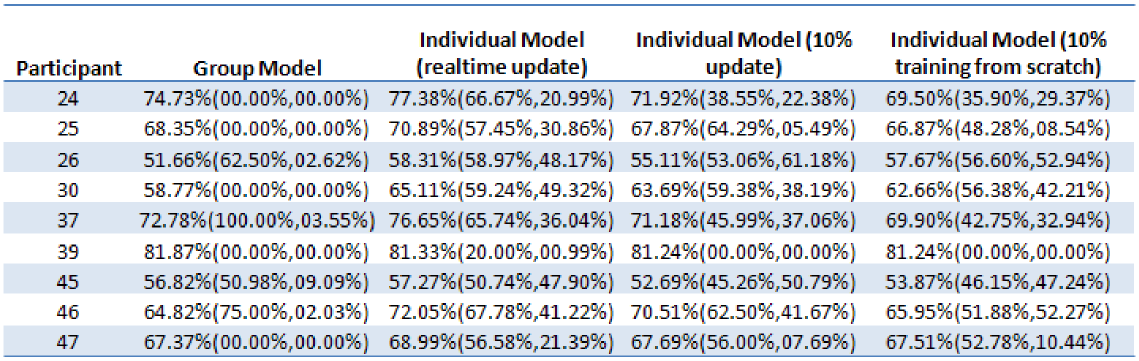
Figure 22: Illustration of change detection and the responses of a participant

Figure 22 illustrates the generation of change detection events in MIX and the responses of a participant, where each block represents a time window and the green diamond represents the response from a participant. There are two types of responses from a participant: correct and incorrect. An incorrect response happens due to timeout, an unnecessary response, or a wrong identification of an event type. We excluded data with an invalid simulation time, data involving threat detection or audio response, and data without change detection events for the models of change detection.

As we did for task demand task analysis, we first tested the model for groups, where the model was trained by 70% of the data. The remaining 30% of the data was used for testing. The accuracy we got for task demand is 68%. The precision and recall for incorrect responses are 59.72% and 29.08%, respectively.

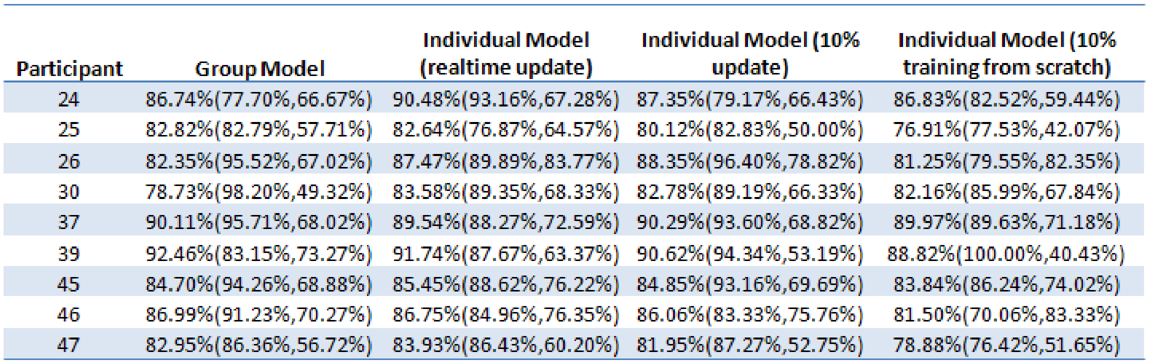
In the second study for change detection, we trained the model using the data from all participants except for 24, 25, 26, 30, 37, 39, 45, 46, and 47. And then we tested the model using the data for participants 24, 25, 26, 30, 37, 39, 45, 46, and 47. The accuracy we got is 67% (Precision 58%: Recall: 2% for incorrect responses).

In the third study for change detection, we compared the accuracy of group models and three flavors of individual models. As shown in Figure 23, the performance of current models for both groups and individuals is not stable, where accuracy can drop to as low as 50% for some participants. One hypothesis is that the attributes used for model development might not be enough to predict the performance of participants for change detection tasks. These attributes include user profiles and physiological sensor data from eye tracking and ECG. One possibility is to include system data to predict the performance of a participant, such as waiting time or response time. The experimental results in Figure 24 confirmed our hypothesis, where we can predict the performance of change detection with accuracy over 78% for group models and 76% for individual models. The results also showed that an individual model built upon a group model was better than the one built from scratch.



Participant	Group Model	Individual Model (realtime update)	Individual Model (10% update)	Individual Model (10% training from scratch)
24	74.73%(00.00%,00.00%)	77.38%(66.67%,20.99%)	71.92%(38.55%,22.38%)	69.50%(35.90%,29.37%)
25	68.35%(00.00%,00.00%)	70.89%(57.45%,30.86%)	67.87%(64.29%,05.49%)	66.87%(48.28%,08.54%)
26	51.66%(62.50%,02.62%)	58.31%(58.97%,48.17%)	55.11%(53.06%,61.18%)	57.67%(56.60%,52.94%)
30	58.77%(00.00%,00.00%)	65.11%(59.24%,49.32%)	63.69%(59.38%,38.19%)	62.66%(56.38%,42.21%)
37	72.78%(100.00%,03.55%)	76.65%(65.74%,36.04%)	71.18%(45.99%,37.06%)	69.90%(42.75%,32.94%)
39	81.87%(00.00%,00.00%)	81.33%(20.00%,00.99%)	81.24%(00.00%,00.00%)	81.24%(00.00%,00.00%)
45	56.82%(50.98%,09.09%)	57.27%(50.74%,47.90%)	52.69%(45.26%,50.79%)	53.87%(46.15%,47.24%)
46	64.82%(75.00%,02.03%)	72.05%(67.78%,41.22%)	70.51%(62.50%,41.67%)	65.95%(51.88%,52.27%)
47	67.37%(00.00%,00.00%)	68.99%(56.58%,21.39%)	67.69%(56.00%,07.69%)	67.51%(52.78%,10.44%)

Figure 23: Experimental results for change detection (without response time)



Participant	Group Model	Individual Model (realtime update)	Individual Model (10% update)	Individual Model (10% training from scratch)
24	86.74%(77.70%,66.67%)	90.48%(93.16%,67.28%)	87.35%(79.17%,66.43%)	86.83%(82.52%,59.44%)
25	82.82%(82.79%,57.71%)	82.64%(76.87%,64.57%)	80.12%(82.83%,50.00%)	76.91%(77.53%,42.07%)
26	82.35%(95.52%,67.02%)	87.47%(89.89%,83.77%)	88.35%(96.40%,78.82%)	81.25%(79.55%,82.35%)
30	78.73%(98.20%,49.32%)	83.58%(89.35%,68.33%)	82.78%(89.19%,66.33%)	82.16%(85.99%,67.84%)
37	90.11%(95.71%,68.02%)	89.54%(88.27%,72.59%)	90.29%(93.60%,68.82%)	89.97%(89.63%,71.18%)
39	92.46%(83.15%,73.27%)	91.74%(87.67%,63.37%)	90.62%(94.34%,53.19%)	88.82%(100.00%,40.43%)
45	84.70%(94.26%,68.88%)	85.45%(88.62%,76.22%)	84.85%(93.16%,69.69%)	83.84%(86.24%,74.02%)
46	86.99%(91.23%,70.27%)	86.75%(84.96%,76.35%)	86.06%(83.33%,75.76%)	81.50%(70.06%,83.33%)
47	82.95%(86.36%,56.72%)	83.93%(86.43%,60.20%)	81.95%(87.27%,52.75%)	78.88%(76.42%,51.65%)

Figure 24: Experimental results for change detection (with response time)

4.4.9 Threat Detection

Threat detection is similar to change detection, where participants were required to identify threats portrayed as armed civilians or enemy soldiers when traveling through the terrain (shown in Figure 25). Similar to model development for change detection tasks, we excluded data with an invalid simulation time, data involving change detection or audio response and data without threat detection events for threat detection (TD).

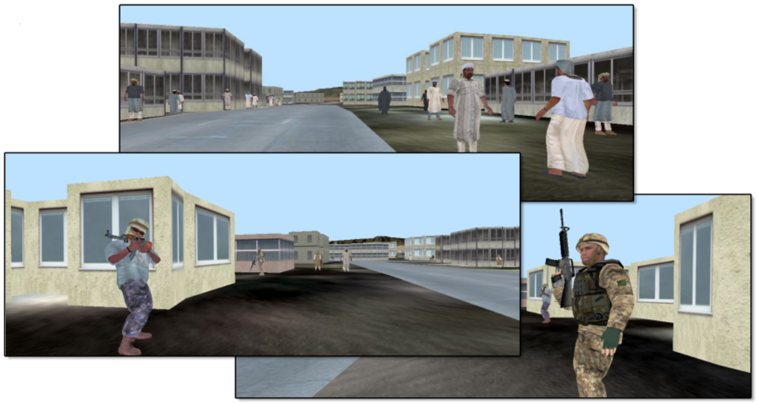
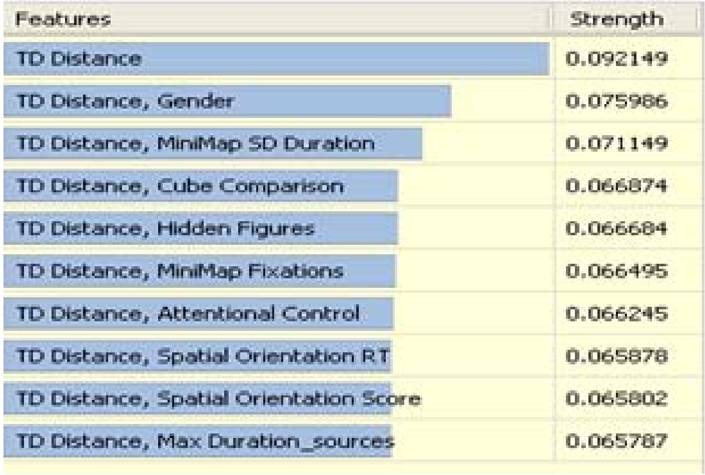


Figure 25: Examples of stimuli present in the threat detection task.

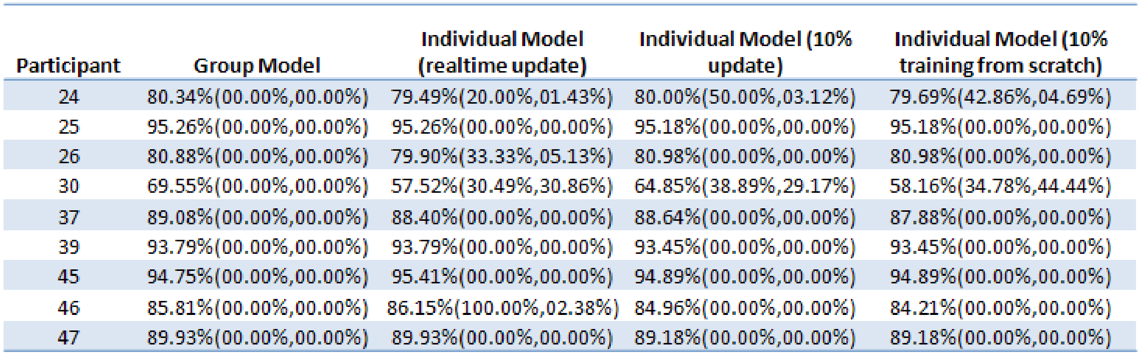


Features	Strength
TD Distance	0.092149
TD Distance, Gender	0.075986
TD Distance, MiniMap SD Duration	0.071149
TD Distance, Cube Comparison	0.066874
TD Distance, Hidden Figures	0.066684
TD Distance, MiniMap Fixations	0.066495
TD Distance, Attentional Control	0.066245
TD Distance, Spatial Orientation RT	0.065878
TD Distance, Spatial Orientation Score	0.065802
TD Distance, Max Duration_sources	0.065787

Figure 26. Experimental results for threat detection.

There are two important variables in the study for threat detection: TD distance (the distance between a threat and the position of a participant in the virtual environment when the participant detects the threat) and TD response time. We found that TD distance was highly correlated with TD response. We tested the model for groups and individuals and found that the accuracy can be as high as 95% when we used TD distance as a single attribute.

Figure 26 displays the experimental results for threat detection, where we excluded both TD distance and TD response time as input attributes for our model. The results show that the models for groups and individuals can predict the performance of threat detection for most of the participants with high accuracy except for participant 30, but the models for groups and individuals cannot predict the incorrect responses of any participants. The precision and recall are close to zero for many participants. This might be caused by the overfitting problem and/or the limitations of the data we have at this time. Different from change detection, there is no triggering event to identify a starting time for threat detection in the existing data set. Therefore, it is impossible for us to define a proper time window, where we can correlate the changes of physiological measurements to threat detection in Bayesian classifiers and develop a real-time cognitive readiness assessment tool. We will discuss the possibility of introducing triggering events for threat detection in future human experiments with the UCF team.



Participant	Group Model	Individual Model (realtime update)	Individual Model (10% update)	Individual Model (10% training from scratch)
24	80.34%(00.00%,00.00%)	79.49%(20.00%,01.43%)	80.00%(50.00%,03.12%)	79.69%(42.86%,04.69%)
25	95.26%(00.00%,00.00%)	95.26%(00.00%,00.00%)	95.18%(00.00%,00.00%)	95.18%(00.00%,00.00%)
26	80.88%(00.00%,00.00%)	79.90%(33.33%,05.13%)	80.98%(00.00%,00.00%)	80.98%(00.00%,00.00%)
30	69.55%(00.00%,00.00%)	57.52%(30.49%,30.86%)	64.85%(38.89%,29.17%)	58.16%(34.78%,44.44%)
37	89.08%(00.00%,00.00%)	88.40%(00.00%,00.00%)	88.64%(00.00%,00.00%)	87.88%(00.00%,00.00%)
39	93.79%(00.00%,00.00%)	93.79%(00.00%,00.00%)	93.45%(00.00%,00.00%)	93.45%(00.00%,00.00%)
45	94.75%(00.00%,00.00%)	95.41%(00.00%,00.00%)	94.89%(00.00%,00.00%)	94.89%(00.00%,00.00%)
46	85.81%(00.00%,00.00%)	86.15%(100.00%,02.38%)	84.96%(00.00%,00.00%)	84.21%(00.00%,00.00%)
47	89.93%(00.00%,00.00%)	89.93%(00.00%,00.00%)	89.18%(00.00%,00.00%)	89.18%(00.00%,00.00%)

Figure 27: Experimental results for threat detection (without TD distance and TD response time)

4.4.10 Multi-dimensional Approach to Cognitive Readiness Assessment

One goal of this project is to develop a reliable, real-time individualized model for cognitive readiness assessment in complex training environments and in the field. Our initial results in the previous reporting period are promising and indicate that it is possible to quickly calibrate the model built for a group of warfighters with limited physiological data and predict overall performance degradation of individual warfighters in a different group. As shown in Figure 28, we combined subjective ratings, user profiles, performance and physiological measures to predict user workload and other metrics of cognitive readiness of Warfighters. The model was used to assess both a component (e.g., a skill) in cognitive readiness and the overall cognitive readiness.

Figure 28: Model development roadmap in Phase I

The quarterly report in the previous period discussed several approaches to cognitive readiness assessment. Two of them are particularly interesting. One is to adapt a group model to individuals in a different group using available training data sets. The other is to learn an individual readiness model from scratch. Generally, an individual readiness model learning from scratch might have a higher accuracy for prediction if there are enough data available. This means we would need a relatively long training battery to access the cognitive readiness of Warfighters. The challenge is that in practice training data sets are extremely small and we need to build a model from the small training data sets. This will greatly shorten the training time for Warfighters, but the risk is that the individual readiness model learning from scratch can be unstable due to limited training data sets.

Alternatively, we can calibrate a group model to assess individual cognitive readiness when the training data sets are small. Figure 29 shows the experimental results for a group model adapted to individuals and individual models learning from scratch when 10% training data sets are used. We can see that the prediction accuracy of task demand, which corresponds to workload or stress in cognitive readiness, are almost the same for group models and individuals. But the group model has a much higher accuracy of change detection compared to individual models. Change detection is related to the pattern recognition skill or perceptual decision making skill in cognitive readiness.

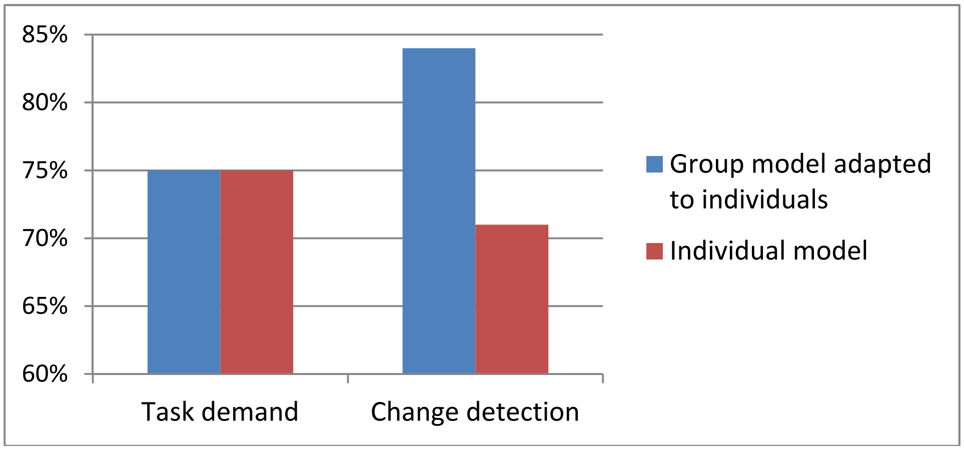


Figure 29: Experimental results for the components of cognitive readiness

In addition to the components for cognitive readiness, we also studied the overall readiness level from subjective, physiological measures and performance measures. Currently, performance measure is a weighted sum of the success rates for all tasks and the weight for each task is assumed to be the same, but the approach can be easily generalized to other scenarios with different priorities of tasks. Three tasks are studied as an example for overall cognitive readiness assessment. These include change detection, threat detection and audio response. Figure 30 shows that the accuracy of our existing model can achieve as high as 86% for overall cognitive readiness assessment. The results indicate that the cognitive readiness assessment model being developed in this project is mature enough for further validation and transitioning to the field.

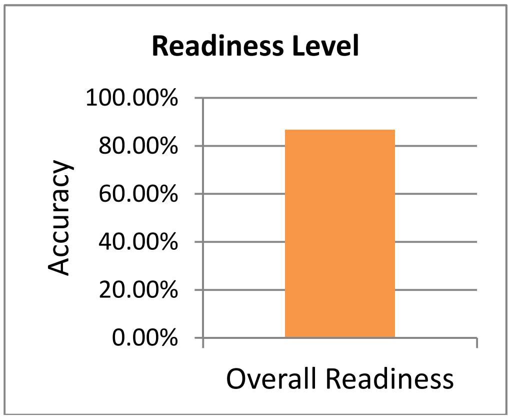


Figure 30: Experimental results for overall cognitive readiness

4.4.11 Discussion

One goal of this project is to develop a reliable, real-time individualized model for cognitive readiness assessment in complex training environments and in the field. Our initial results are promising and indicate that it is possible to quickly calibrate the model built for the groups of warfighters with limited physiological data and predict overall performance degradation of individual warfighters. However, it is relatively difficult to predict the performance of warfighters for some tasks such as change detection and threat detection using eye tracking and ECG data. These two tasks usually need instant feedback and are highly variable compared to task demand in a multi-tasking environment. Also, there is not enough training data for incorrect responses of both threat detection and change detection. This may cause an overfitting problem for the AODE classifier. It will be interesting to see if we can get a similar conclusion when we integrate EEG data with ECG and eye tracking data.